

DOCUMENT RESUME

ED 175 446

IR 007 613

TITLE Renaissance Man: The Key Component. Association for Educational Data Systems Proceedings: 17th Annual Convention, May 14-18, 1979.

INSTITUTION Association for Educational Data Systems, Washington, D.C.

PUB DATE May 79

NOTE 378p.

AVAILABLE FROM Association for Educational Data Systems, 1201 Sixteenth Street, N.W., Washington, D.C. 20036 (\$10.00 prepaid)

EDRS PRICE MF01 Plus Postage. PC Not Available from EDRS.

DESCRIPTORS *Computer Assisted Instruction; *Computer Managed Instruction; *Computers; Computer Science Education; *Electronic Data Processing; Elementary Secondary Education; Higher Education; Management

IDENTIFIERS *Computer Centers

ABSTRACT

Eighty-six papers presented at the 17th Annual Association for Educational Data Systems convention cover the educational application of computers in the following areas: computer assisted instruction, instructional and learning processes, computer-related curriculum, educational administration, computer resources, and data-center administration. Many of the papers are introduced by author abstracts and contain research methods, results, and lists of references cited. For ease in locating individual papers, separate title and author indexes are provided. (RAO)

* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *

ED175446

U.S. DEPARTMENT OF HEALTH,
EDUCATION & WELFARE
NATIONAL INSTITUTE OF
EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY.

Association For Educational Data Systems

AEDS

PROCEEDINGS

17th ANNUAL CONVENTION

**RENAISSANCE MAN:
THE KEY COMPONENT**

**MAY 14-18, 1979
Renaissance Center
Detroit, Michigan**

"PERMISSION TO REPRODUCE THIS
MATERIAL IN MICROFICHE ONLY
HAS BEEN GRANTED BY

Shirley Easterwood

TO THE EDUCATIONAL RESOURCES
INFORMATION CENTER (ERIC)."

IR007613

PREFACE

In keeping with the AEDS-'79 Convention theme "Renaissance Man: The Key Component" it is our hope that these proceedings will give you, the "key component", a "re-birth" in the use of computers and data processing in Education.

This publication is the result of many hours of hard work on the part of the authors, their secretaries, the AEDS-'79 Program Committee, and the Publications Committee. These proceedings should make an excellent supplement to the information received from attendance at the respective sessions. The papers contained within these proceedings have been compiled from many that were submitted to our AEDS-'79 Program Committee. Each session speaker was asked to submit a summary of his or her presentation in a form ready for publication using a specific format and type font. Eighty-six of the approximately one-hundred presentations are represented within these proceedings; most of the others did not respond to the request for manuscripts. A few papers were omitted because they did not conform to the requested format and even if we were to have re-typed them, the illustrations contained within them were such that they would not have photo-copied satisfactorily. We sincerely hope that these authors will understand why their papers are not included in these proceedings.

The AEDS-'79 Convention Team is deeply grateful to each contributor for his or her participation in these proceedings. It is our desire that this collection of papers will serve to promote and advance the understanding of educational data processing within the educational community and that this publication will be of assistance to you as a valuable reference guide in the years ahead.

Lary R. Smith
For the AEDS-'79 Convention Team

Note: In putting together this convention, we have had many comments about the use of the sexist term "Man" in our convention theme. Please understand that its use here is to imply "mankind" in the general form and not a specific sex. We hope you will accept this apology for our seeming "male chauvinistic" attitude.

FORWARD

The Association for Educational Data Systems (AEDS) is a private, non-profit international educational organization founded in 1962 by a group of professional educators and technical specialists in the educational application of computers and technology. The aim of the organization is to provide a forum for the exchange of ideas and information about the relationship of modern technology to education.

AEDS membership is open to all persons interested in learning more about and keeping informed of current developments and directions in educational data systems and computer technology. Included in the membership are educators and technical specialists from all over the United States, Canada, and other countries representing public and private schools, higher education, professional education, state and provincial departments of education, federal government, the military, and other professional and technical groups interested in the educational use of computers and technology.

AEDS-'79, the Association's 17th Annual Convention, is designed to bring together professionals who are interested in learning and sharing information about the latest developments in the application of technology to the educational process. These proceedings represent a significant contribution to the growing body of knowledge in this vital area. The AEDS-'79 Convention Team is grateful to the many contributors to these proceedings. The Convention Team firmly believes this collection of papers can have considerable impact upon the educational community if all participants give each paper the careful attention it is due.

The AEDS-'79 Convention Team would like to express its appreciation for the support and assistance of all of those who have shown an interest in this Convention. The MAEDS (Michigan Association for Educational Data Systems) and MACUL (Michigan Association for Computer Users in Learning) Boards of Directors have been most supportive, and many members of the AEDS-'79 Convention Team are active members or even Directors of MAEDS and/or MACUL. We also have received guidance from all of our co-hosting organizations relative to potential exhibitors, sessions, activities, and fees. The MAEDS and MACUL Boards of Directors, the AEDS Board of Directors, the other co-hosting organizations, their memberships, and others have exhibited great interest, initiative, and support, all of which are essential to making any convention a success.

Most especially, I would like to thank all of the members of the AEDS-'79 Convention Team who have worked so tirelessly on the many details associated with this Convention.

Arthur W. Daniels, Jr.
Convention Coordinator

TABLE OF CONTENTS

INTRODUCTION

Preface	iii
Forward	v
AEDS Board of Directors	xiii
AEDS-'79 Convention Team	xiii
MAEDS Board of Directors	xiv
MACUL Board of Directors	xiv
Other AEDS-'79 Co-Hosts	xv

COMPUTER-ASSISTED INSTRUCTION

The Effects of Computer-Assisted-Instruction on Low Achieving Math Students in the Junior High School - Ione T Goodman (Friday am) . . .	3
The Effects of Supplementary CAI on the Math and Reading Achievement of EMR High School Students - A Thomas Vincent (Thursday am)	6
English Strands: A Status Report - Bruce G Alcock, David Nicholson & Starr Snead (Friday am)	12
English Usage Exercises II: A CAI Series for the Elementary School - John G Allee & Peter J Quinn (Wednesday pm)	15
Exploring the Potential Use of Computer Assisted Instruction in a Nursing Curriculum - Patricia J Morin & Julia Ann Fisco (Wednesday pm)	19
Humanism for the Computer Age in Education: Planning for Man/Machine Synchronicity - George Donoian (Tuesday pm)	21
Magnet Math - Drawing the Best from Your Students by Computer - Betty Vanden Bosch (Thursday pm)	27
Micro in a Macro World: Curriculum Development for the Microcomputer - Shelley Yorke Rose (Wednesday pm)	33
"Universite du Quebec" PLATO System - Paul-Andre Desjardins (Friday am)	37
The Word-Problem Strand in Math Drill Using a Mini-System - G Lawrence Hopkins (Wednesday am)	39

INSTRUCTION/LEARNING APPLICATIONS

AUMS: A Score Language for Computer-Assisted Applications in Music - W Kenton Bales & Joan C Groom (Friday am)	47
--	----

Automatic Programming: An Educator's Aid For Computer-Assisted Test Construction (CATC) - Ronald W Collins & Steven J Duff (Wednesday am)	51
Computer Access to Information: The Academic Library of Tomorrow - Robert R Korfhage & Christine L Borgman (Tuesday pm)	58
Computer Assisted Test Construction at the University of Georgia - John S Edwards & James E Eisele (Tuesday pm)	62
The Computer Attitude Continuum (CAC): Sorting the Critics and the Appreciators - Walter M Mathews & Abraham W Wolf (Wednesday pm) . .	66
Computer-Based Instruction in Medical Education - William E Sorlie & Diane L Essex (Friday am)	70
Computer Management of Educational Competency Statements - Fred G Smith & John S Edwards (Tuesday pm)	74
A Computer Management System for a Modularized Course with Hierarchical Structure and Optional Units - Ann Brebner, J.H. Hallworth & W.D. Winn (Tuesday pm)	76
Computers in the College Classroom: An Example from Transportation Geography - C Murray Austin (Friday am).	81
Contrasts in Achievement of Comp Skills by Low and High Ability 6th Grade Students in 1965 SMSG & 1975 Modern Elem. Math Programs - Ann D Hungerman (Wednesday pm)	85
CRSIE - A Computer Resource System for Instructional Evaluation - Randy Rousse (Wednesday pm)	88
CVIS: Where Career Education and Computer Meet - Dianne K Garnett & Robert F Garnett (Wednesday am)	91
Educational Administration, Evaluation, Research and Package Programs: An Integrated EDP Application - S James Zafirau (Thursday pm) . . .	95
Effects of Computer-Based Instruction and Tutoring in an Elementary School System - Lotus M Knief (Wednesday am)	98
Elementary Computers - James Hollingsworth (Wednesday am)	102
File-Management and Formatting Techniques for an APL Test Generation System - Daniel R Lawrence (Thursday am)	104
Instructional Decision Support Systems: Design Considerations - William C. Bozeman (Wednesday pm)	109
The Interaction of Heuristic/Algorithmic Instructions and Conceptual Preference in Computer Based Problem Solving - Joan M Sustik & Bobby R Brown (Friday am)	114
Interactive Computer-Based Exercises in Radiology - Kreps, Kraft, Burris & Olson (Wednesday pm)	119
Random Generator for Arithmetic Expression Problems - James D Jensen (Thursday pm)	125
The Significance of Uncontrolled Variables on Research Outcomes of Computer Assisted Instruction - J F Huntington (Friday am)	127
The Taming of a MINIMICROMAX - Leighton A Price & Raymond J Kalush, Jr (Thursday pm)	130
The T.I.E.S. Instructional Management Support System - Donald Holznagel & James Sydow (Thursday pm)	136

The TRS-80 as a Classroom Teaching Tool: Teaching Hebrew and Religious Studies - Samuel W Spero (Wednesday am)	140
Use of Information Retrieval Programs from A (IDS to Job Hunting) to Z (EBRA Keys) - Janice M. Glime (Thursday pm)	144
Use of the Personal Computer as an In-class Teaching Tool in Introductory Business Statistics - Sidney P Sytsma (Thursday pm) . .	150
Using the Computer to Foster Creative Interaction Among Students - Sharon Dugdale (Wednesday pm)	153
Uses of Computer Simulations in Introductory Psychology - Douglas B Eamon & William N Hayes (Thursday am)	160
Utilizing the Computer to Teach and Study Clinical Problem Solving Skills - Doron Gil, Chris Wagner & John Vinsonhaler (Tuesday pm) . .	164
"We Deliver" (Jobs) - Robert J Miller, Gerri Powell-Rae & Roland L Meade (Wednesday am)	169

COMPUTER-RELATED CURRICULUM

The Assessment of Students' Computer Literacy -- Results from the MECC Computer Literacy Study - Anderson, Hansen, Johnson & Klassen (Wednesday am)	175
Breaking Down the Barriers: The Real Micro Miracle - Darlene E Crawford (Tuesday pm)	177
CAI Literacy in Instructional Technology Curriculum - Marsha Young (Thursday am)	181
The Computer in Cartoons: The Saturday Review, 1946-1976 - Kaye Reifers & Walter M Mathews (Thursday am)	184
Computer Literacy for 4th, 5th & 6th Graders - Sam Ewing (Wednesday pm)	188
A Computer Literacy Module for Junior High School Students - Ione Goodman, Steve Wetherbee, Daniel Neaton & John Marshak (Thursday pm)	191
Computer Literacy, School Mathematics, and Problem Solving (3 for the price of 1) - Fred Bell (Thursday pm)	194
A Computer Supported Field Based Teacher Preparation Program - John H Schnabel (Wednesday pm)	198
Educational Programs in Computers and Society and Computer Literacy - Richard H Austing & Gerald L Engel (Tuesday pm)	203
Experiences of "My Pal the Computer" in the Primary Grades - Jean Rice & Sandy O'Connor (Wednesday am)	210
An In-Service Model to Prepare All Elementary Teachers in One School District to Use Computers - Gerard J Putz (Thursday pm)	213
Integrating Computer Concepts into Community College Business Curricula - Roy James Parrish, Jr. (Wednesday pm)	215
An Interdisciplinary Approach to Computer Science - William W Agresti (Friday am)	219
The Introduction of CAI to an Undergraduate Decision Making Course - Susan S Lenker & Edward J Fisher (Friday am)	222

The Projects with Industry Approach to Training and Placing Disabled Students as Computer Programmers - Robert J Leneway (Wednesday pm)	225
There are no Blackboards - Ronald V Carswell (Wednesday am)	228

EDUCATION ADMINISTRATIVE APPLICATIONS

The Computer Based Hourly Classroom Attendance System in Kalamazoo Public Schools - Jerry Fochtman (Wednesday am)	235
A Data Management System for a Small Liberal Arts College - Victor A Meyer & Ignatia Frye (Tuesday pm)	238
Data Processing in the Year Round 45/15 Plan School District - Donald J Vinlove (Wednesday pm)	244
A Degree Progress Audit of Students by Computer - James E Greene & Lester A Singletary (Tuesday pm)	247
Delaware Technical & Community College Student Record System-Why it is Different - Frank A Kros III (Thursday am)	252
A Generalized Management System for Educational Data - John S Edwards & Pier Paolo Pisani (Tuesday pm)	256
Projecting Enrollments for Elementary and Secondary Schools - William J Hierstein (Wednesday pm)	259
SCORE: A Test Scoring and Analysis Program - Edwin J Jaufmann & Robert L Williams (Wednesday pm)	262
A Systematic Information Management Model for Cooperative Education: Data Base Development & Evaluation - Robin D. Avery & Jeanne Dulaney-Sorochak (Thursday pm)	266
Using a Computer to Aid in Evaluating the Occup. Activities of a Student's Progress in a Physician's Assistant Program - Edwards, Gallagher, Valgora, Campbell & Costello (Wednesday am)	270

COMPUTER RESOURCES

CAMIL and the Effective Implementation of Instructional Software for a Large Scale CMI/CAI System - David Pflasterer & Anne Montgomery (Tuesday pm)	277
Humanistic Technology: Toward a Renaissance for Education - D Thomas King (Tuesday pm)	283
Instructional and Research Considerations with an Intelligent Videodisc - Bobby Brown, Lawrence Stolurow, Barbara Fowler & Joan Sustik (Wednesday pm)	287
Logical Distributed Data Processing for School Districts - S M Raucher (Wednesday am)	291
The Microprocessor as Pencil - John D'Angelo (Tuesday pm)	295
Random Access Audio Hardware, Software and Courseware - Donald H McClain (Wednesday am)	308
Selecting Microcomputers for the Classroom - David B Thomas & Donald H McClain (Wednesday pm)	312
Teachers Taking a Byte from an Apple - Earl Keyser (Friday am)	319

DATA-CENTER ADMINISTRATION

The Case of an Educational Data Center's Environmental Transition: Its Affect on People, Places, and Things - Philip A. Mackey (Friday am)	325
Choosing a Classroom Microcomputer - Ed Herstein (Thursday pm)	328
The Development of a Management Information System: A Case Study - Edwin D Bell (Wednesday am)	330
The Dialectic of Artist and Technologist in Instructional Systems Development - G.P. Kearsley & A.F. O'Neal (Thursday pm)	333
Documentation Standards and Their Application to Computer Systems - Frank B Thomas (Thursday pm)	336
Establishing a CAI Center and Developing Software: Problems, Concerns, and Recommendations - Thomas S Jordan (Friday am)	341
Florida's Strategy for Optimizing Computing Resources for Educational Administration and Instruction - M D Roblyer & David Brittain (Tuesday pm)	346
From Timesharing to Microcomputers - Donald C Holznagel (Wednesday am)	351
Implementation of the Apple II Microcomputer in Minnesota - Linda Borry & John Arneson (Thursday pm)	356
Institutional Planners and Computer Center Administrators: Need for Greater Dialogue - Robert F Garnett & Dianne Garnett (Thursday pm)	359
Maintenance Problems: Do It Yourself And You May Do It Better - Lawrence J Molloy & David B Preston (Wednesday am)	363
A Market Needs Approach to the Systematic Selection of an Economically Efficient Educational Computer Resources System - Joseph L Braden & Asad S Khailany (Tuesday pm)	367

INDEX

Alphabetical by Title	371
Alphabetical by Author	375

ACKNOWLEDGEMENTS

AEDS BOARD OF DIRECTORS (1978-79)

PRESIDENT

Brad Burris
Scott Foresman & Co.
1900 E. Lake Ave.
Glenview, IL 60025

PRESIDENT-ELECT

Don Holznagel
TIES
1925 W. County Rd B-2
Roseville, MN 55113

PAST PRESIDENT

Judith Edwards
Alaska Telecommunications
526 Main Street
Juneau, AK 99801

SECRETARY

Sister Patricia Marshall (1980)
Xavier University
7325 Palmetto Street
New Orleans, LA 70125

TREASURER

Delbert Repp (1979)
Office of Computer Services
2 Pleasant Avenue West
Lancaster, NY 14086

DIRECTORS

Winston Addis (1980)
Superintendent of Schools
Mt. Vernon Community Schools
Mt. Vernon, IA 52314

Bruce Alcock (1981)

Riverdale Country School
S. Fieldston Rd., W. 253 St.
Bronx, NY 10471

Arthur Daniels, Jr. (ex officio)
RAMS Cooperative
31201 Dorchester
Madison Heights, MI 48071

John Hamblen (1979)
University of Missouri at Rolla
Rolla, MO 65401

DIRECTORS

D. Thomas King (1980)
St. Paul Public Schools
360 Colborne Street
St. Paul, MN 55102

Marvin Miller (1981)
Lincoln Public Schools
P.O. Box 82889
Lincoln, NE 68501

Paul Pair (1979)
Control Data Corp.
1650 W. Alameda Dr.
Tempe, AZ 85282

Dennis Spuck (ex officio)
University of Houston
4800 Calhoun
Houston, TX 77004

EXECUTIVE SECRETARY
Shirley Easterwood
1201 16th Street N.W.
Washington, D.C. 20036

AEDS-'79 CONVENTION TEAM

CONVENTION COORDINATOR

Arthur W. Daniels, Jr.
RAMS Cooperative
31201 Dorchester
Madison Heights, MI 48071

ASST. CONVENTION COORDINATOR

Lary R. Smith
Wayne County ISD
33500 Van Born Road
Wayne, MI 48184

ACTIVITIES - CANADA

Harry/Honi Groves
Windsor Board of Education
451 Park Street West
Windsor, Ontario, Canada N9A-6K1

ACTIVITIES - UNITED STATES

James Smith
PRIME Computers, Inc.
4150 Clifford
Brighton, MI 48116

EQUIPMENT

Tim Spanke/Duane Miller
Pontiac City Schools
44 State Street
Pontiac, MI 48053

EVALUATIONS

John Hamblen
University of Missouri at Rolla
Rolla, MO 65401

EXHIBITS

George Kargilis
Warren Consolidated Schools
29900 Lorraine
Warren, MI 48093

FACILITIES

Richard Alverson
Dearborn Public Schools
2024 Carlsyle
Dearborn, MI 48124

FINANCE

Tom Schmeltzer
Oakland Schools
2100 Pontiac Lake Road
Pontiac, MI 48054

PROGRAM

Don Mann, CDP
Oakland Community College
27055 Orchard Lake Road
Farmington Hills, MI 48018

PUBLICITY

Jack Daniels/Jerry Lamphere
Detroit College of Business
4801 Oakman Blvd.
Dearborn, MI 48124

**MICHIGAN ASSOCIATION FOR EDUCATIONAL DATA SYSTEMS (MAEDS)
BOARD OF DIRECTORS (1978-79)**

PRESIDENT

Joan Rathbun
Oakland Schools
2100 Pontiac Lake Road
Pontiac, MI 48054

PRESIDENT-ELECT

Don Mann, CDP
Oakland Community College
27055 Orchard Lake Road
Farmington Hills, MI 48018

PAST-PRESIDENT

George Kargilis
Warren Consolidated Schools
29900 Lorraine Blvd.
Warren, MI 48093

SECRETARY

Lary R. Smith (1980)
Wayne County ISD
33500 Van Born Road
Wayne, MI 48184

TREASURER

Dave Rhodes (1979)
Berrien ISD
711 St. Joseph Ave.
Berrien Springs, MI 49103

DIRECTORS

Debbie Belaire (1979)
Van Buren Public Schools
555 West Columbia
Belleville, MI 48111

Nicholas Colovos (1980)
Lansing Public Schools
519 West Kalamazoo
Lansing, MI 48933

Arthur W. Daniels, Jr. (1981)
RAMS Cooperative
31201 Dorchester
Madison Heights, MI 48071

DIRECTORS

David Ginsburg (1979)
Oak Park School District
13900 Granzon
Oak Park, MI 48237

Larry Molloy (1979)
Wayne County ISD
33500 Van Born Road
Wayne, MI 48184

John Skewes (1980)
Oakland Schools
2100 Pontiac Lake Road
Pontiac, MI 48054

DATA VIEWS EDITOR

Mel Staebler
Pontiac School District
44 State Street
Pontiac, MI 48053

**MICHIGAN ASSOCIATION FOR COMPUTERS USERS IN LEARNING (MACUL)
BOARD OF DIRECTORS (1978-79)**

PRESIDENT

Leighton A. Price
500 Computer Center
Michigan State University
East Lansing, MI 48824

PRESIDENT-ELECT

Tom Schmeltzer
Oakland Schools
2100 Pontiac Lake Road
Pontiac, MI 48054

PAST-PRESIDENT

Larry Molloy
Wayne County ISD
33500 Van Born Road
Wayne, MI 48184

SECRETARY

Ray Hoag
Grand Rapids Jr. College
143 Bostwick Avenue N.E.
Grand Rapids, MI 49503

TREASURER

Richard Alverson
Dearborn Public Schools
2024 Carlisle
Dearborn, MI 48124

COMMUNICATIONS SECRETARY

Lary R. Smith
Wayne County ISD
33500 Van Born Road
Wayne, MI 48184

JOURNAL EDITOR

Roland Meade
Washtenaw ISD
P.O. Box 1406
Ann Arbor, MI 48106

DIRECTORS

Paul Brimo (1979)
East Detroit High School
15501 Couzens
East Detroit, MI 48021

Margaret V. Nielsen (1979)
Genesee ISD
2413 West Maple Avenue
Flint, MI 48507

Jo Ellen Toroian (1980)
Regional Data Processing Center
2630 West Howell Road
Mason, MI 48854

Jim Winebrenner (1980)
Livonia Public Schools
15125 Farmington Road
Livonia, MI 48154

OTHER AEDS-'79 CO-HOSTING ORGANIZATIONS

Mich Assoc of Intermed. School Administrators
Hugh H. Tayler
President, MAISA
Muskegon Area ISD
630 Harvey Street
Muskegon, MI 49442

Michigan Assoc for Media in Education
Elaine Didier
Executive Secretary, MAME
Bureau of School Services
401 S. Fourth Street
Ann Arbor, MI 48103

Michigan Assoc of School Administrators
Donald M. Currie
Executive Secretary, MASA
421 West Kalamazoo
Lansing, MI 48933

Michigan Assoc of Secondary School Principals
Robert Rodda
Associate Director, MASSP
Bureau of School Services
401 S. Fourth Street
Ann Arbor, MI 48103

Michigan Educational Research Assoc
William Veitch
President, MERA
Oakland Schools
2100 Pontiac Lake Road
Pontiac, MI 48054

Michigan Intermediate Media Assoc
George Grimes
President, MIDMA
Wayne County ISD
33500 Van Born
Wayne, MI 48184

Ontario Assoc for School Business Officials
W.J. McCordic
Executive Secretary, OASBO
Suite N-1201
252 Bloor Street West
Toronto, Ontario M5G-1V5

SouthEastern Michigan Computer Organization
Jim Rarus
President, SEMCO
Warren Consolidated Schools
29900 Lorraine
Warren, MI 48093

COMPUTER - ASSISTED INSTRUCTION

THE EFFECTS OF COMPUTER-ASSISTED-INSTRUCTION
ON LOW ACHIEVING MATH STUDENTS IN THE JUNIOR HIGH SCHOOL

Ione T. Goodman
Ann Arbor Public Schools

An NDEA title III, a grant through the General Education Services of Michigan Department of Education, enabled a one year research project to be conducted at Tappan Int. School in Ann Arbor, Michigan. Students were included who were found to be in the lower 1 and 2 stanines on eighth graders C.A.T. tests and/or on seventh graders who were in the lower 20% mastery on seventh grade MEAP tests. Ten minute daily participation in the CAI arithmetic drill and practice units on the Hewlett Packard \$M2 programs resulted in improvement in all concepts presented in CAI program. Fifty percent of the class was black, 20% of the class were handicapped, and 6% were severely handicapped.

The idea of computer assisted instruction (CAI) is nearly twenty years old.

A terminal which looks like a TV or a typewriter asks individual students questions, gets their answers, and grades them on the spot.

But computer assisted instruction is expensive. The only reason Tappan Intermediate School in Ann Arbor, Michigan was able to afford it was because of an NDEA title III grant through the General Services of the Michigan Department of Education for a one year research project for low achieving junior high students.

CAI works, and it is especially effective with students who are bored by traditional teaching methods.

These students are in math classes because they are suppose to be in math classes. They're doing math because they're suppose to be doing math. They're not really very excited about it. Reaching these children is always a problem. The good students are easy to reach. They're always glad to do more work. But it's the child that isn't too enthusiastic that's hard to reach. I have seen students refuse to work on their traditional math lessons but insist upon having two sessions in one day on a computer terminal. I have had students who couldn't read but after two weeks on a CAI reading program they were eagerly reading a math text book and understanding what they read. I have watched students progress from multiplying by two's or three's to dividing by a two digit numeral in a few months and enjoy doing it. Severely handicapped students have done their lessons on the computer terminals as easily as their physically capable peers.

Students on the math CAI benefit a great deal from the reading in the math program because they begin to realize that if they do not read the information correctly, they will not get the next step and they will have to go back and repeat it, or they will sit there with nothing happening. These children are excited about getting on the terminal and they don't want to leave. They want supervision while on the termi-

nal. Some students get very angry at the terminal rather than the teacher. But they will come to the teacher to complain about the terminal and to seek help in doing their computer math. Many teachers have feared that computer instruction would replace teachers. Students are more insistent in getting help from a teacher than when they are taught in a traditional math class.

The Tappan project focused on those students in seventh and eighth grade who were identified as deficient in basic mathematic skills. Eighth graders who were in the project were in the lower 20% mastery on seventh grade MEAP tests and seventh graders were there because they scored low on a 6th grade math skills test. Students were not randomly selected for entrance into the project. They entered the project voluntarily with their parents approval.

These students were put on the Stanford math drill-and-practice program: \$M2, for 10 minutes each day. This was taken as a daily supplementary math lesson for they also had a daily 50 minute math class.

The \$M2 math program has the "strands" approach. A strand is a series of problems involving the same operation. Operations are ordered according to their difficulty. Each lesson includes problems from several strands. Students usually attempted between 20 and 30 problems presented by a computer terminal that linked the school with a computer at the Washtenaw Intermediate School District. Ten minutes was allowed for each lesson and they had a twenty seconds response time.

The CAI program is highly individualized. Each student received problems that are based on his past performance. The math drill and practice curricula is composed of blocks which are similar to chapters in a book. Blocks are assembled into groups of 24 and each group is referred to as a year. Each block is composed of a pretest, five main lessons, and a post test. Each of the five main lessons have five levels of difficulty. At the beginning of each block,

4 the student takes a pretest. His score on the pretest determines the level of difficulty at which the material will be presented to him in the first main lesson in the block. At the end of each block, the student takes a post-test. Any block for which the student scored less than 85% of the post-test is scheduled for review.

The \$M2 course offers an accelerated review of basic computational skills in junior or senior high who have computational deficiencies which prevent them from performing effectively in other courses. Those concepts normally covered in grades one through three are contained in year 1 of this curriculum. Topics from grade four through six are covered in year 2.

Design

A one-group pretest-posttest design was used in the study. The dependent variable was test scores from a sixth grade basic math skills test. The independent variable, or treatment, was the CAI math drill-and-practice program.

Subjects

The sample consisted of 15 students from the 7th and 8th grades of Tappan Intermediate School in Ann Arbor, Michigan. Most of the students came from low-income families. There were seven 7th graders and eight 8th graders. Ten were females and five were males. Seven were black and eight were white Caucasians. Two students were physically handicapped and one had emotional problems.

Instruments

Ann Arbor Sixth Grade Mathematics Assessment Test (AAGG): This is a sixty question multiple choice test on math basic skills designed by Ann Arbor sixth grade teachers. This was used as a pre and post test.

Goodman's Math Attitude Test (GOATT): This instrument has twenty statements. Items are rated on a five point scale. The contents of the statements are based on Bloom's Taxonomy - Affective Domain. This opinionaire was used pre and post.

Seventh Grade Michigan Assessment Program (7MEAP): Only the seventh graders took this as a pre and post test. They did the math part which had 225 questions.

Eighth Grade California Achievement Test (8CAT): Only eighth graders took the 63 question math sections as a pre and post test.

CAI Tests

The pre and post tests for each block in the \$M2 math program are built into the CAI unit. The CAI curriculums keep daily records and cumulative records of each student's progress.

Computer

The computer was a Hewlett-Packard 2000F. Both CRT terminals and hard copy terminals were used.

Reading Results

Two students who were in the project that could not read were put on a computerized reading program: \$RD for 10 minutes twice a week. The Woodcock Reading Assessment (oral) was made pre and post CAI reading instruction on one eighth grade girl.

<u>Date</u>	<u>Letter Ident</u>	<u>Word Ident</u>	<u>Word Attach</u>	<u>Word Comp</u>	<u>Passage Comp</u>
10-76	4.3	2.0	1.6	2.6	2.4
6-78	12.9	2.9	2.2	7.1	4.3

The Gates Reading Assessment (written) was made on the other eighth grade girl pre and post CAI Reading instruction

<u>Date</u>	<u>Vocab</u>	<u>Comp</u>
9-76	3.2	3.1
5-78	4.9	3.7

Findings and Comments

1. A statistical test, a paired t-test, was made on the pre and post sixth grade test scores. There was a significant difference in their scores at $\alpha = .01$ with 12 degree of freedom.
2. Students were performing two to four years below grade level in arithmetic when entering the project. Two of these students were able to perform at their regular grade level the following year.
3. A student's present level of performance in a variety of math problems is quickly located; and he is paced according to his progress in these problems.
4. The cost of a remedial specialist versus the cost of a computer-aided remedial program should be taken into account in determining the relative efficiency of the two methods. If an elementary math or reading CAI program is proved to be an efficient form of remedial instruction, then the need for remedial specialists could be considerably reduced.

PRE AND POST TEST SCORES

	Pre AAOG	Post AAOG	Pre GOATT	Post GOATT	Pre 7MEAP	Post 7MEAP	Pre 8CAT	Post 8CAT	Male	Female
8th Graders										
1	26	33	72	68					X	
2	17	30	71	57			22	40		X
3		32		69			34	40	X	
4	0	19	64	61					X	
5	18	28	57	57						X
6	20	36	52	53			30	29		X
7	20	20	68	49						X
8	18	26	87	92						X
7th Graders										
1	27	36	69	94	181	190				X
2	31	32	58	70	164	197				X
3	17	22			87	86			X	
4	36	42	82	60	184	204				X
5	15	31	65	53	79	108				X
6	31	30	77	80	172	167				X
7	16		48						X	

COMPUTER TEST SCORES

Tappan School

#M2 Course History Report

Regular Students

Year 1

Block	# Stu.	Pre-Test Avg. S.D.		Post-Test Avg. S.D.		Increase Avg. S.D.	
1	24	86.0	8.3	93.1	5.3	6.4	10.2
2	21	77.4	13.6	88.9	9.6	11.5	14.3
3	20	89.2	12.0	95.8	6.1	6.7	13.1
4	18	88.6	8.9	91.1	8.9	2.5	11.0
5	17	70.0	19.7	76.8	15.9	6.8	15.0
6	16	86.0	8.7	89.1	9.6	3.2	10.6
7	16	80.2	14.2	86.5	11.2	6.3	12.6
8	16	81.5	14.2	87.2	12.2	5.6	14.1
9	17	90.4	5.7	92.5	8.2	2.0	8.7
10	15	66.4	16.1	70.6	18.6	4.1	8.0
11	15	83.7	13.4	89.3	10.4	5.6	10.0
12	15	89.3	11.2	98.3	2.4	9.0	11.4
13	15	88.0	6.8	89.9	7.5	1.8	8.7
14	14	90.3	8.4	94.9	8.3	4.6	9.6
15	14	85.1	6.4	89.6	7.9	4.5	11.6
16	14	66.1	8.5	72.9	10.7	6.8	9.5
17	11	88.9	6.1	93.9	3.0	5.1	5.4
18	11	87.0	8.1	90.5	5.4	3.5	8.4
19	9	91.1	6.5	95.3	2.5	4.2	5.7
20	9	81.9	10.9	86.2	12.8	4.4	10.3
21	8	61.9	20.3	81.9	9.7	20.0	22.8
22	8	89.6	5.4	94.3	5.6	4.7	5.5
23	8	74.2	17.8	81.3	9.9	7.0	17.5
24	7	91.0	3.0	94.1	2.2	3.1	4.0

THE EFFECTS OF SUPPLEMENTARY CAI ON THE MATH AND READING ACHIEVEMENT OF EMR HIGH SCHOOL STUDENTS

A. Thomas Vincent, Ed. D.
Belmont High School, Dayton, OH

The implementation of PL 94-142 will create a demand for an individualized curriculum for each educable mentally retarded student. Meaningful CAI programs can assist teachers in accomplishing the goals and objectives of their IEPs.

The studies outlined in this paper show that the academic math and reading skills of secondary EMR students can be significantly improved through participation in a supplemental CAI program. Furthermore, the students attitude toward math and reading can be significantly improved through CAI.

INTRODUCTION

Two similar empirical studies were conducted. The first, in 1976, was a supplemental CAI reading program. The second involved CAI mathematics and was implemented in 1977. Both studies were designed for EMR students.

Due to space restrictions, this paper speaks directly to the mathematics study. The reader may assume that the design for the reading study was nearly identical to the math program.

PURPOSE AND RATIONALE

The first purpose was to compare the cognitive mathematics skills of secondary Educable Mentally Retarded (EMR) students who participated in a supplemental Computer Assisted Instruction (CAI) mathematics program with those students who did not participate in such a program. The second, to determine if the students' participation in the CAI mathematics program would result in a more positive attitude toward mathematics. Expectations were that the school attended would not have a significant effect on achievement or attitude toward mathematics.

The problem has significant to society for educators, parents, and courts are concerned with the quality of education for all students. Recently the courts have decreed that all students are entitled to a quality education in the least restrictive environment. This law will mandate mainstreaming some students who have been called educably retarded, deaf, learning disabled, et al.

Teachers and administrators may be able to use data generated by this study to improve the instruction in the high school EMR classes. To design new teaching strategies, CAI could be useful as a

supplemental teaching tool for EMR students when they are placed in the regular high school classes.

The academic progress of students with functional or motivational deficiencies tends to compound itself. As the student continues in school, there is a tendency to a greater total academic lag. Robinson and Robinson state that "for these children (EMR) the rate of intellectual development is usually no more than one-half or three-fourths that of the average child."¹

It can be established that the EMR student is physically of high school age but educationally functional at the elementary school level. This student, while having difficulty with academic skills, does, in fact, respond to the same teaching techniques as 'normal' students do. It is important then that effective teaching techniques involving CAI be studied in an attempt to improve the mathematics achievement level of EMR students.

The need for drill and practice is clear; the burden on the classroom teacher to provide and evaluate the materials is also clear. Norris supports the role of the computer to relieve the classroom teacher of this tedious task when, in relating to special education, he states: "The next area where CAI will become cost-effective is in special education, which today is very costly. It is also highly frustrating and time-consuming for teachers. The individual testing, prescribing, feedback, record keeping and analyzing--all done manually--require too much time, effort, and patience to be done effectively. The computer has infinite patience. It doesn't neglect any of those chores in favor of something more interesting. And it is private. The pacing factor in

getting widespread usage is availability of courseware."²

It is not the contention of this paper that computers can do the whole job by replacing the classroom teacher. A sophisticated CAI program can assist the teacher with diagnosis, prescription, instruction, and evaluation. However, as Suydam so clearly points out, there are human needs to consider. She says, "CAI is unable to do the whole teaching job, obviously. There is still the need for interaction among learners, whether they be 7 or 17 or 70. The computer does not (at least yet) react to the human needs of the child. The teacher must take care of these. The computer may greet the child by name but it does not compliment him on his new shoes."³

POPULATION

The EMR population of two metropolitan high schools were the subjects. Thirty-one students were randomly assigned to the experimental group and thirty-nine to the control group. The subjects were males and females, blacks and whites, in grades nine through twelve. There was no division of students by experimental and control group for classroom instruction.

PROGRAM AND PROCEDURE

The CAI mathematics curriculum was developed at Stanford University. Hewlett Packard Corporation adapted this program to the 2000C Model computer. The system maintained and printed out extensive information concerning each student's individual progress.

The Wide Range Achievement Test (WRAT) was administered as a pretest.⁴ Scores of the experimental group were used for placement at the proper grade level on the computer program. The teachers used a diagnostic prescription model which encouraged individual pace through the curriculum. As part of class instruction time, students were sent to the terminal room to participate in the CAI program for a period of ten minutes per day, three to four days per week.

The study was conducted for ten weeks. The WRAT was used as the post-test instrument. Students also completed the Spikerman Mathematics Attitude Instrument.⁵

HYPOTHESES AND RESULTS

It was hypothesized that EMR students who participated in the supplemental CAI mathematics drill and practice program would score significantly higher in basic mathematics skills than the students who did not participate in the program. There would be no significant interaction (school by treatment) on achievement. An analysis of covariance using

the pretest scores as covariates, was employed. It was found that the difference between the groups on their adjusted means was significant ($\alpha = .05$). (Appendix A) School attended was not a significant factor and there was no treatment by school interaction.

It was also hypothesized that students who participated in the CAI drill and practice program would demonstrate a significantly more positive attitude toward mathematics than those students who did not participate in the treatment. Also, there would be no significant interaction (school by treatment) on attitude.

Results of an analysis of variance indicate that those students who participated in the CAI program did have a significantly ($\alpha = .05$) more positive attitude toward mathematics than those not participating in the program. (Appendix B) School attended was not a significant factor in the attitude gain; there was no significant interaction.

A two way ANCOVA showed that there was not a significant race, sex or grade level main effect. Neither did any of these have an interaction effect on the significant achievement made by the experimental group over the control group. A correlation coefficient between time spent on the computer and achievement was not significant.

In the study dealing with CAI reading, similar results were established. To compute the statistical data, the Biomed P2V analysis of variance and covariance, developed by the University of California was used. The difference between the experimental and control group on their adjusted means for comprehension was significant ($\alpha = .05$). (Appendix C) There was no significant difference in vocabulary.

Attitude toward reading was measured as part of the post-test analysis. The students who participated in the CAI supplemental reading program had a significantly ($\alpha = .05$) more positive attitude toward reading than those in the control group. (Appendix D)

CONCLUSIONS

It was concluded that the cognitive academic mathematical skills of the EMR students who participated in the supplemental CAI mathematics and reading programs were improved significantly more than the skills of students who did not participate in the program. In terms of attitude toward mathematics and reading, the students in the CAI programs had a more positive attitude toward mathematics and reading than the students who were not in the program. School attended was not a factor in achievement or attitude formation.

Analysis of achievement in relation to demographic characteristics indicated that none of these had a significant effect. In this study, the amount of time students

8 spent on the terminals had no significant effect. In this study, the amount of time students spent on the terminals had no significant effect on achievement.

It can be stated that there is only a five percent possibility that the greater achievement and more positive attitude of the experimental group was caused by chance alone. It can also be stated that the school the students attended, the sex, race, grade level of the students, and the amount of time they spent working on the terminal had no effect on the achievement gain or more positive attitude of the experimental group.

OBSERVATIONS

Due to the number of subjects available for the studies, no group was established to control for the Hawthorne effect. Therefore, there were factors which may have contributed to the greater achievement gains and more positive attitude demonstrated by the experimental group. Knowing that they were part of the group that participated in the CAI program, the students may have exerted extra effort to achieve. The fact that the terminal rooms were remote to the classroom and necessitated the students moving from one room to another may have been a motivating force. Since the CAI experience was part of the classroom time, the release time from the regular classroom could have generated greater interest and motivation to succeed. When students work with the computer, there is a certain excitement and enthusiasm that is generally observable. These factors, plus the novelty of the computer experience, may have been contributing factors.

The EMR mathematics resource teachers commented that generally the students in the experimental group "looked forward to going to the terminal room". Students expressed pride in being trusted to move through the building without supervision.

The classroom teachers spent more time with the students in the experimental group. The generally positive reinforcement that the teachers gave the students when reading their print outs may have been a contributing factor, or even the total factor which caused their greater achievement.

It is believed that other factors may have prevented even greater achievement by the experimental group. The computer mathematics program was written for general use by elementary students. As a result, students in the experimental group encountered concepts that had not been presented in the classroom. This problem caused frustration until the classroom teacher could explain the concept. Some students became upset to the point that they expressed a desire not to continue the CAI lessons. The resource teacher's patient counseling overcame this

feeling.

Since the study was started in March, teacher orientation lasted only one day. This was not considered adequate time for a teacher inservice training. Printed material needs to be made available that would give the teachers a clear understanding of the concepts the students would encounter lesson by lesson and block by block. It was observed that for some students these factors may have actually caused a reduction in academic achievement and positive attitude change.

At School B, the computer was not available for instructional purposes after the study was completed. At School A, the computer was available until the end of the school year. At that school, the students in the experimental group were told that it was not necessary for them to continue their computer lessons after the post-test, but if they wanted to, they could go on with the lessons until the end of the year. Twelve of the sixteen students in the experimental group decided to continue the lessons on a voluntary basis. This supported the predominantly positive comments that the students made during the ten week study period.

There were spin offs for students in the experimental group that were observed in an unobtrusive way. The fact that the students had to move from one part of the building to another to take the computer lesson developed a responsibility for personal actions and accountability for time. Going to the computer room gave students the opportunity to interact with students from classes other than their own. The EMR student comments were in the tone that this made them feel more a part of the entire school.

In some instances, concepts the students had experienced and worked with in class were presented in a different way by the computer. This afforded the teacher the opportunity to reinforce the concept through two different explanations.

The sophistication of the hardware and software contributed to the process as well as the product skills of the students. EMR resource teachers stated that students often do addition, subtraction, and multiplication starting in the left hand column and working to the right hand column. With the CAI program used in this study, the typing cylinder is always positioned under the column to be calculated, i.e. right to left. Each column of addition, subtraction or multiplication is checked for accuracy before the cylinder moves to the next column. This rigid procedure reinforces the process used in these three concepts.

The change in attitude of the EMR staff toward the CAI program was dramatic. When the concept of EMR students working on the computer was first presented, the staff was reluctant to participate. Some of their comments were:

1. "The students cannot operate the terminals."

2. "The students do not know how to type."
3. "The students cannot be trusted in the halls to get to the terminal room and back to class on time."
4. "We cannot work out a schedule to get them there."
5. "They (the students) won't know how to work all the problems."

After weeks of discussion and examination of the literature which indicated that EMR students could function with the hardware and grow cognitively from the experience, the staff, with some hesitation, was willing to participate in the program.

By the time the study was completed, there was a general consensus of enthusiasm by the EMR staff for the CAI experience. Some of their comments were:

1. "I had no idea the students would like it (CAI) that much."
2. "I can't believe how well the students get to the terminal room and back."
3. "I wonder if we can do a study to see why some students did so much better than others?"
4. "Can we have the terminals next year?"

The EMR staff recommended to the Special Education Department that the hardware and software be made available for all EMR students to participate in the supplemental drill and practice mathematics program. They also recommended that a supplemental CAI reading program be made available to all EMR students.

RECOMMENDATIONS FOR FURTHER STUDIES

To minimize student frustration and to maximize the coordination of the academic effort, a computer curriculum should be established that would be closely correlated with the curriculum used in the classroom. If this were accomplished, it would assist the classroom teacher in individualizing the program. The teacher would no longer need to prepare two tracts--one for the classroom curriculum and one for the material presented by the computer. The concept presentations could also be more closely aligned for the student. The problem of students encountering concepts that are not part of the classroom curriculum could be controlled.

A larger sampling of secondary EMR students should be established so that a control group for the Hawthorne effect could be established. This group would have computer experience but would not participate in the mathematics lessons. Their experience would consist of participating in

another program which would be presented on the same hardware. This would enable the researcher to establish if the novelty of the computer experience is a contributing factor to the greater cognitive gains made by the experimental group in this study.

This study found that the amount of time the students spent working with the computer was not a significant factor in achievement. This finding may be due to the fact that the total amount of time the students spent was relatively similar. A study with a larger population could vary the total amount of time the students spend on the lessons and also vary the number of days per week the students take CAI lessons. The results could give educators insight to determine the optimum time for different students. If possible, the design could identify the students with their attitude scores. This would enable the researcher to determine if there is a significant linear correlation between attitude and achievement.

From the literature, there is reason to believe that if meaningful CAI curriculum were written in other disciplines, results similar to those found in this study should be established to determine if significant achievement and attitude gains can be made with elementary grade level EMR students using CAI in various disciplines.

RECOMMENDATIONS FOR IMPLEMENTATION

While there is need for additional research, the results of this study indicate that program implementation could occur at this time. Before a CAI supplemental program is started, it is critical that it be established that the software on the computer is of a quality that will promote the goals and objectives of the school and teachers. As pointed out in the literature, software that is written by a person with little or no experience in the field may produce results short of expectations. Furthermore, when a quality program has been obtained in any discipline, it must be placed on hardware that has the sophistication to allow for the full use of the curriculum, as well as the capability for extensive record keeping of student progress.

Another key to the success of the program is the willingness of the teachers involved to participate in and work with the supplemental drill and practice program. Once a willing staff has been established, then a carefully planned inservice must be carried out. Such an inservice program should be conducted when the hardware and software of the CAI program is functional. The resource teachers could then participate in the CAI program on the computer as the students would. The teachers should learn to interpret the print out of the students progress as they work through the program and under-

10 stand how this information can be used to strengthen an individualized classroom model. This time would give the staff an opportunity to study and discuss printed material that would offer a full explanation of the concepts in the program.

If these criteria are met, then educators may anticipate that when secondary EMR students participate in a supplemental drill and practice CAI mathematics and reading program, their cognitive academic skills will be significantly improved over students who do not participate in such programs. Also, the students will develop a more positive attitude toward mathematics and reading than students who do not have the CAI experience.

REFERENCES

¹Halbert R. Robinson and Nancy M. Robinson, The Mentally Retarded Child (New York: McGraw-Hill, 1965), p. 271.

²William C. Norris, "Via Technology to a New Era in Education," Phi Delta Kappan, Vol. 58, No. 6, February, 1977, p. 452.

³William G. Cunningham, "The Need for Dialogue Between Educators and Technologists," Phi Delta Kappan, Vol. 58, No. 8, February, 1977, p. 450.

⁴J. F. Jastak and S. R. Jastak, WRAT Manual, The Wide Range Achievement Test (Wilmington, Guidance Associates, 1965), p. 13.

⁵Spikerman, "A Study of the Relationship between Attitudes toward Mathematics and some Selected Pupil Characteristics in a Kentucky High School," p. 57.

If a full bibliography is desired, the author may be contacted at Belmont High School, 2323 Maplevue Avenue, Dayton, OH 45420 and a copy will be sent.

APPENDIX A
FACTORIAL ANALYSIS OF COVARIANCE FOR ACHIEVEMENT

SOURCE OF VARIATION	df	MS	F	Probability less than
Treatment (A)	1	1.874	6.186	0.016
School (B)	1	0.152	0.502	0.481
A x B	1	0.125	0.081	0.776
Error	65	3.301		

APPENDIX B
FACTORIAL ANALYSIS OF VARIANCE FOR ATTITUDE

SOURCE OF VARIATION	df	MS	F	Probability less than
Treatment (A)	1	548.171	10.070	0.002
School (B)	1	17.372	0.319	0.574
A x B	1	1.219	0.022	0.882
Error				

APPENDIX C
COVARIANT TEST OF COMPREHENSION

SOURCE OF ERROR	df	MS	F	p	S.S.
Between group Exp. vs. control	1	4.37	5.17	.032*	4.37
Within group	25	.845	-	**	21.12

* p .05
**p .01

APPENDIX D
READING ATTITUDE TEST

t score = 2.76	D.F. = 26	P less than .01
----------------	-----------	-----------------

ENGLISH STRANDS: A STATUS REPORT

Bruce G. Alcock
Riverdale Country School
David Nicholson
Riverdale Country School
Starr Snead
Greenwich Country Day School

This paper describes the development work being done to produce a series of English drill programs that will help the teacher by providing supplementary drill work without requiring extra work on the part of the teacher. The programs use the strands concept of curriculum organization and keep records on student progress.

Since 1973, Riverdale Country School has been involved in the development and implementation of educational software, specifically in the area of computer assisted instruction (CAI). With a complete mathematics drill package for the elementary grades up and running successfully, the perfect complement is a comparable system in English; this is now in the process of being developed after a long and difficult start.

Five years ago, Riverdale personnel began writing education software. Working in conjunction with teachers from other schools, a series of math drill programs based on the curriculum that Patrick Suppes developed at Stanford University was begun. The approach that he used took arithmetic and divided it into thirteen topics called strands (e.g. addition, subtraction, multiplication, etc.). In the summer of 1974 Michael Fulop, who was entering the tenth grade, wrote the first version of the Math strands. They were a series of separate programs but appeared to the student as one continuous flow. Different strands were randomly chosen and problems of the appropriate level within the strand were printed out. The student had two chances to answer the question correctly but if he missed the first try, the problem was marked as being done incorrectly for the records. These records were used in succeeding sessions to generate problems at the appropriate level in each strand. Through a year of use by several hundred students, difficulties and inadequacies were identified. The following summer, the same student made revisions to the Math strands. One problem that had existed was the inability for the teacher to enroll students while strand sessions were being run. This was a terrible inconvenience since students were doing their math drills at all times of the day (one of the schools using the Math strands was a boarding school). The revisions included a new file structure that permitted

enrolling to be done while students were running sessions.

Prompted by the success of the math package, the same student was commissioned to begin work on English. A vocabulary program was written, as well as a general purpose data manipulation program. While the latter is currently still in use, especially for entering curriculum for the new version of English, the vocabulary program has not been used due to the voluminous amount of data required; each word has seven drills, a definition, an example, and a hint. The instructional strategies seem quite sound, but require a great deal of work for even a small set of vocabulary words to be implemented. In 1976 more work was done in English but again only programs are the result; since no data has been entered in the past two years, the programs are currently useless at this time. Work was begun by David Solomon, who was a student entering the tenth grade, on revising the Math strands to handle up to 128 terminals on one mini-computer while still permitting the system to do other tasks. During the year he incorporated curriculum revisions and additions made by G. Lawrence Hopkins at the Masters School into the new strands program. In the summer of 1977 work again was done on Math and English but still no final product emerged.

During the spring of 1978 planning was begun for a new effort into producing a workable English program that would not require overwhelming effort. Because of the success of the strands, it was felt that the strand concept should be adhered to, as well as produce real material that English teachers would find useful. Some of the lessons that we learned from the Math strands and that we are attempting to create in English strands are: ease of use, fairly simple but comprehensive curriculum and scores kept automatically by the computer.

With the help of a grant from the Helena Rubinstein Foundation, and the school's Parents Association, Riverdale prepared to start a dual pronged attack on the English. Two students, Phil Terman and Eric Leventhal, were engaged to help design and do the programming on the project. The English teachers came from Riverdale and Greenwich Country Day Schools.

As mentioned above, the math curriculum was adapted from an existing curriculum outline and, therefore, needed only minimal revisions to become operational. The English program, however, did not have any similar counterpart; thus work was begun without any real guide. In writing the curriculum, we worked at two levels, aiming at students in the fourth-through-sixth grade range and at students in the ninth-through-eleventh grade range.

The topics for the high school program were organized into six strands: (1) Punctuation, (2) Syntax, (3) Grammar and Usage, (4) Diction and Style, (5) Spelling, and (6) Vocabulary. Of these, we worked on only the first two this summer, with the intention of completing parts of the program which could then be put to trial use immediately. To keep the curriculum simple to write and easy for the students to use, a standard grammar-text format was adopted, in which the computer presents a rule stating the concept to be drilled, then gives a set of examples to be worked on. If the student performs satisfactorily, he moves on to the next rule and its drills; if not, he must go through another set of drills on the rule he has not mastered. A variety of drill formats were found useful. The punctuation exercises invite the student to distinguish between complete and incomplete sentences, to insert commas, periods, semicolons, and other marks in the appropriate places, to divide words into syllables for hyphenation, to punctuate direct discourse, and so on, as in a standard school grammar. The syntax drills, including work on subordination, placement of modifiers, and sentence completeness, make use of a multiple-choice format in which the student is asked to correct, rewrite, or combine sentences and choose the correct response, after which he is given the complete rewritten sentence (instant feedback). If the punctuation and syntax drills prove successful as designed, the

The program is intended to supplement but not to replace the equivalent work in the classroom. Bright students might conceivably teach themselves using this curriculum, but we have not thought of it as a "programmed text". The curriculum will make it possible for the teacher to teach in class and save the onerous drill work for the students' session on the computer. Nor will the program take the place of essay writing and correcting. Many points of grammar and style are too subtle to be mastered without a live teacher responding to the student's written work. Like any grammar study, the program will perhaps make the task of learning to write a little easier by getting certain preliminary matters out of the way.

We have tried to finesse the problem of which theory of grammar to use (traditional vs. transformational-generative) by relying on common-sense and operational definitions in writing the rules. Standard grammars of both kinds were consulted, but the results are probably more traditional than otherwise. In any case, the emphasis is on the drill work and not on the rules and definitions, which we have included essentially to organize the work and to guide those using it. We have tried to make the curriculum as generally useful as possible.

In developing Language Arts curriculum at the fourth through sixth grade levels, a paragraph format has been devised. Starr Sneed, a fifth-grade teacher who is also a specialist in Reading and in Learning Disabilities, and Phil Sneed, a School Psychologist and former fifth-and-sixth grade teacher, are incorporating a highly flexible battery of skills exercises into each of the fourth, fifth, and sixth-grade levels. The basis of the curriculum is analysis of several elementary school Language Arts texts.

The paragraph format adds a fair measure of complexity to the task of writing material; but it is felt that the extra effort will pay off in higher level of reader interest, and thus in stronger motivation for these younger children to utilize the program. Each paragraph can be presented in any one of eighty-one permutations, thus accommodating a broad range of individual differences. A pupil's progress through this curriculum

will be determined by consecutive-successes criterion measures.

Five or six sentences are printed out in paragraph style and then questions are asked on each one of the sentences. Each question covers a different English strand, e.g. vocabulary, spelling, punctuation and usage. Some sentences in the paragraph are filler sentences on which no questions are asked. If a student is weak in one area, it is possible to specify one strand and then questions are only asked in that area. To avoid making curriculum development too complex, drills are drawn from those used in the paragraph's construction.

High school teachers seem to prefer concentration on more specific skills and, therefore, no paragraphs are used; rather emphasis is placed on college-board-type drills. Three types of approaches are available: assorted drills in various strands, drills in only one strand topic or specify a particular skill in one strand. The latter is emphasized as being required by high school teachers. A list is made available to each teacher with a computer identification code and a brief name for the skills.

General drill in English, however, is still available for the student who is weak, who simply wants to review basic English skills or who wants to prepare for a competency exam or the college boards. In the four years that we have researched and developed English material, we have run across no program that provides the student with exercises in more than one skill in English. Most English texts also seem to fragment the topics. We feel this approach is important since skills are used jointly, especially in writing a paper.

In developing the curriculum, an overall pattern was laid out within which more specific outlining was done. From this we could tell approximately how many strands would be necessary, but work has been restricted to only a small group. While we eventually hope to have a complete curriculum, we are working a strand at a time in the tenth grade and grade level at a time with four strands on the fifth grade level. In this way the job becomes easier to manage and to establish checkpoints at which a working system, even though not complete, exists which students are able to use. Within these small and manageable areas, the

curriculum has been broken down into greater detail and drills developed. After a year of testing through use of the program in conjunction with regular classwork, we will be able to determine what revisions are necessary and how effective our overall approach has been.

Both the English and Math programs are very easy for the student to run. All that need be typed is "ENGLISH" (or "MATH"), whereupon the computer asks for the student's name and number. The student's record is retrieved by the computer and used for presenting drill material at the appropriate level. At the end of the session, the student's record is updated and a brief score summary is displayed. The student does not have to learn any login procedures, but only how to type one or two simple commands, type return at the end of a line, and use rubout or delete to correct a typing error. Modifiers to English and Math can be used to specify a topic but these are also quite simple, e.g. ENGLISH VOCAB would give drills in vocabulary. Next summer work will be necessary to expand the curriculum on both levels as well as incorporate feedback from teachers using the package. As the task has become more clearly defined and work has begun, enthusiasm by teachers has begun to grow. Even teachers who previously were quite hesitant are now interested in learning to use the material developed during the summer and seem to be quite willing to experiment. The Math strands were successful because they represented the first product that teachers could use as an adjunct to the classwork with ease. This theme has been stressed in the English package as well.

The major problem in preparing for the summer effort has been finding funding, as well as willing and capable curriculum developers. Programming work has always been done by the students and, therefore, presents no major difficulty. Programmers are always available and willing to work on new projects such as this one.

Developing an integrated package is an ambitious project but one which is long overdue. Having had success with a Math program, the staff at the Riverdale Computer Center is now attempting to tackle the next step in providing needed educational software that will help make the computer an integrated part of the modern school.

ENGLISH USAGE EXERCISES II:
A CAI Series for the Elementary School

John G. Allee
The George Washington University

Peter J. Quinn
Robert L. Williams
St. John's University, New York

ENGLISH USAGE EXERCISES II (EUX II) is a set of 102 computer terminal sessions that explore grammatical relationships that often pose written usage problems for many students, the intended audience being, as for EUX I (96 sessions), the intermediate grades. Each session is composed of a short didactic lesson and a set of sentence exercises, each designed for a sitting of from 10 to 12 minutes. Forty DTSS BASIC programs generate the sessions, each stand-alone program presenting from 2 to 10 lessons. Easily transported, EUX II, like EUX I, is available to schools for the cost of handling.

EUX II sessions, like those of its predecessor, EUX I, are written in Dartmouth BASIC, and may be run on either cathode ray tube or conventional terminals.1/ The mini-lessons, sentence exercises, questions, and comments of EUX II are composed from relatively limited data statements within stand-alone programs.

A session of EUX II follows:

EUX251

AND 1 OR 1

TYPE YOUR FIRST NAME. ? JACK

WHICH OF THE 4 SESSIONS OF 'AND 1 OR 1' ARE YOU
TO DO? 1 2 3 OR 4 ? 1

JACK: LET'S STUDY SOME SENTENCES!

>>> TOM IS TEASING JILL. <<<<
READY ?

'TOM' IS A SINGLE SUBJECT THAT COUNTS ONE.
READY ?

'IS' COUNTS ONE.
READY ?

>>> SUE IS TEASING JILL. <<<<
READY ?

'SUE' IS A SINGLE SUBJECT THAT COUNTS ONE.
READY ?

'IS' COUNTS ONE.
READY ?

>>> TOM AND SUE ARE TEASING JILL.
READY ?

'AND' CONNECTS THE TWO SINGLE SUBJECTS.
READY ?

WHILE CONNECTING THEM, 'AND' ADDS TWO SINGLE SUBJECTS TO COUNT 2.
READY ?

'ARE' AGREES BY COUNTING 2.
READY ?

>>> TOM OR SUE IS TEASING JILL.
READY ?

'OR' CONNECTS THE TWO SINGLE SUBJECTS.
READY ?

WHILE CONNECTING THEM, 'OR' CHOOSES ONE SINGLE SUBJECT TO COUNT 1.
READY ?

'IS' AGREES BY COUNTING 1.
READY ?

>> 'AND' CONNECTS AND ADDS TWO.
>> 'OR' CONNECTS AND CHOOSES ONE.
READY ?

JACK: LET'S CHOOSE HELPING VERBS THAT COUNT AND AGREE
WITH SUBJECTS.

1. SUE OR TIM ---- TEASING KAREN. (IS,ARE) ? 0
NO, JACK!
A. OR CHOOSES ONE SINGLE SUBJECT TO COUNT 1.
B. THE HELPING VERB 'IS' COUNTS 1.

2. ---- ANN AND BOB SCOLDING ROD. (IS,ARE) ? 0
NO, JACK!
A. AND ADDS TWO SINGLE SUBJECTS TO COUNT 2.
B. THE HELPING VERB 'ARE' COUNTS 2 OR MORE.

3. ---- JOE AND MARY PRAISING CATHY? (IS,ARE) ? 0
NO, JACK!
A. AND ADDS TWO SINGLE SUBJECTS TO COUNT 2.
B. THE HELPING VERB 'ARE' COUNTS 2 OR MORE.

4. ---- TIM AIDING ALICE? (IS,ARE) ? 0
NO, JACK!
A. ONE SINGLE SUBJECT COUNTS 1.
B. THE HELPING VERB 'IS' COUNTS 1.

5. BOB OR KAREN ---- HELPING BILL. (IS,ARE) ? 0
NO, JACK!
A. OR CHOOSES ONE SINGLE SUBJECT TO COUNT 1.
B. THE HELPING VERB 'IS' COUNTS 1.

6. MARY AND ROD ---- SELECTING JIM. (IS,ARE) ? 0
NO, JACK!
A. AND ADDS TWO SINGLE SUBJECTS TO COUNT 2.
B. THE HELPING VERB 'ARE' COUNTS 2 OR MORE.

7. ---- JILL OR CATHY IGNORING KAREN? (IS,ARE) ? 0
NO, JACK!
A. OR CHOOSES ONE SINGLE SUBJECT TO COUNT 1.
B. THE HELPING VERB 'IS' COUNTS 1.

8. ---- KAREN OR ALICE AVOIDING ROD? (IS,ARE) ? 0
NO, JACK!
A. OR CHOOSES ONE SINGLE SUBJECT TO COUNT 1.
B. THE HELPING VERB 'IS' COUNTS 1.

9. ---- ROD AND BILL DISTURBING CATHY? (IS,ARE) ? 0
NO, JACK!
A. AND ADDS TWO SINGLE SUBJECTS TO COUNT 2.
B. THE HELPING VERB 'ARE' COUNTS 2 OR MORE.

10. CATHY ---- REBUKING ALICE. (IS,ARE) ? 0
NO, JACK!
A. ONE SINGLE SUBJECT COUNTS 1.
B. THE HELPING VERB 'IS' COUNTS 1.

This session is the first in a series of eight that introduces coordinating conjunctions, the forty-ninth lesson of EUX II. Following a short mini-lesson, a student using EUX is usually presented a set of sentence exercises. In most instances, the student spells one word responses, like IS or ARE in this lesson.

A few lessons are totally explanatory, or didactic. The first session of the 'A & THE' series, EUX263, follows:

EUX263

A & THE III

TYPE YOUR FIRST NAME. ? ALICE

'A & THE III' HAS 2 LESSONS. WHICH ARE YOU TO DO? 1 OR 2 ? 1

LET'S LOOK AT 'PEAR':

A PEAR TWO PEARS SEVERAL PEARS

READY ?

LET'S LOOK AT 'SUGAR':

A SUGAR TWO SUGARS SEVERAL SUGARS

READY ?

THESE PHRASES WITH 'SUGAR' ARE NOT ENGLISH. WE NORMALLY WOULDN'T USE THESE, ALICE.

READY ?

WE CAN COUNT 'PEARS.' BUT WE CAN'T COUNT 'SUGAR.'

LET'S LOOK AT 'BOOK':

A BOOK TWO BOOKS SEVERAL BOOKS

READY ?

LET'S LOOK AT 'WATER':

A WATER TWO WATERS SEVERAL WATERS

READY ?

THESE PHRASES WITH 'WATER' ARE NOT ENGLISH. WE NORMALLY WOULDN'T USE THESE, ALICE.

READY ?

WE CAN COUNT 'BOOKS.' BUT WE CAN'T COUNT 'WATER.'

READY ?

COUNTABLES UNCOUNTABLES

PEARS	SUGAR
BOOKS	WATERS

READY ?

ALICE, LET'S SEE HOW WE DO COUNT UNCOUNTABLES.

COUNTABLES UNCOUNTABLES

A LUMP < OF < SUGAR
TWO LUMPS < OF < SUGAR

READY ?

A CUP < OF < WATER
TWO CUPS < OF < WATER

READY ?

'A' MARKS ONLY COUNTABLES.

READY ?

'A' CAN'T MARK UNCOUNTABLES.

READY ?

'A' COUNTS ONE.

READY ?

ALICE, HERE ARE SOME EXAMPLE SENTENCES:

SUE IS EATING A PEAR.

READY ?

JOE IS TASTING A LUMP OF SUGAR.

READY ?

ANN IS READING TWO BOOKS.

READY ?

ROD IS DRINKING TWO CUPS OF WATER.

READY ?

BOB IS EATING A SLICE OF BREAD.

READY ?

GOODBYE, ALICE.

This series, which follows the series of the earlier example--'AND & OR'--commences a carefully sequenced approach to some intricacies of language not usually treated in standard language arts textbooks. In EUX I, significant attention was given to establishing verb and auxiliary number, but the subjects of exercise sentences of that volume were kept relatively simple. The 'AND & OR' series, for example, with its requests for verb responses assumes the student has been well-drilled on the auxiliaries previously, it being expected that he should be able to concentrate on AND and OR because they influence the 'counts' of compound subjects, as in this section of EUX251, session 4:

EUX251

AND & OR I

TYPE YOUR FIRST NAME. ? JACK

WHICH OF THE 4 SESSIONS OF 'AND & OR I' ARE YOU TO DO? 1 2 3 OR 4 ? 4

JACK, LET'S STUDY SOME SENTENCES:

>>> TOM AND SUE HAVE SCOLDED JILL.

READY ?

'AND' CONNECTS THE TWO SINGLE SUBJECTS.

READY ?

WHILE CONNECTING THEM, 'AND' ADDS TWO SINGLE SUBJECTS TO COUNT 2.

READY ?

'HAVE' AGREES BY COUNTING 2.

READY ?

>>> TOM OR SUE HAS SCOLDED JILL.

READY ?

'OR' CONNECTS THE TWO SINGLE SUBJECTS.

READY ?

WHILE CONNECTING THEM, 'OR' CHOOSES ONE SINGLE SUBJECT TO COUNT 1.

READY ?

'HAS' AGREES BY COUNTING 1.

READY ?

88 >> 'AND' CONNECTS AND ADDS TWO.

88 >> 'OR' CONNECTS AND CHOOSES ONE.

READY ?

JACK, LET'S CHOOSE HELPING VERBS THAT COUNT AND AGREE WITH SUBJECTS.

1. ---- JILL OR CATHY SCOLDED AL? (HAS,HAVE) ? 8
NO, JACK!

A. OR CHOOSES ONE SINGLE SUBJECT TO COUNT 1.
B. THE HELPING VERB 'HAS' COUNTS 1.

2. ---- KAREN AND ALICE PRAISE SAH? (DOES,DO) ? 8
NO, JACK!

A. AND ADDS TWO SINGLE SUBJECTS TO COUNT 2.
B. THE HELPING VERB 'DO' COUNTS 2 OR MORE.

3. ROD AND BILL ---- AIDING RUTH. (IS,ARE) ? 8
NO, JACK!

A. AND ADDS TWO SINGLE SUBJECTS TO COUNT 2.
B. THE HELPING VERB 'ARE' COUNTS 2 OR MORE.

While this example shows compounded proper nouns, the didactic lesson of EUX263 discusses countable (and uncountable common) nouns, the appearance of which demands attention to 0 and S endings. EUX II, in its treatment of subjects, attempts to give a step-by-step, developmental presentation of them, in a manner suggested by the following outline, it to be kept in mind that the series is designed for the intermediate level--grades 4, 5, and 6:

Joe is talking.
Joe and Sue are talking.
Joe or Sue is talking.

He is talking.
He and she are talking.
He or she is talking.
They are talking.

A boy is talking.
The boy is talking.

Two boys are talking.
The boys are talking.

One of the boys is talking.
Both of the boys are talking.

One of the boys and Sue are talking.
One of the boys and she are talking.
One of the boys with Sue is talking.
One of the boys with her is talking.

The last four example sentences suggest the attention EUX II must give conjunctions like AND and OR and prepositions like OF and WITH.

The eleven topics of EUX II and the number of sessions for each follow:

Series	Topics	Number of Lessons
22	N Forms of Verbs	16
23	DN Forms of Verbs	12
24	Spellings Regular Verbs	10
25	Questions	10
26	And & Or	8
27	A & The	4
28	Connectors	4
29	O & S Markers	9
30	Countable Nouns	5
31	Pronouns & Connectors	11
32	Possessives	11

Thus, EUX II delivers 102 sessions, which follow and extend the concepts presented by the 96 lessons of EUX I.

Most of the BASIC Programs that compose EUX II present two or more sessions or lessons. For example, the one program EUX223, "DN Forms of Verbs III," gives five practice sessions:

EUX223

DN FORMS OF VERBS III

TYPE YOUR FIRST NAME. ? PETER

'DN FORMS OF VERBS III' HAS 5 SESSIONS.
WHICH ARE YOU TO STUDY: 1 2 3 4 5 ? 5

PETER, IN THIS SESSION WE'LL OBSERVE THESE VERBS:

O	S	D	DN	IND
CHOOSE	CHOOSSES	CHOSE	CHOSEN	CHOOSING
LAUGH	LAUGHS	LAUGHED	LAUGHED	LAUGHING
SPEAK	SPEAKS	SPOKE	SPOKEN	SPEAKING
CHEER	CHEERS	CHEERED	CHEERED	CHEERING
FORGET	FORGETS	FORGOT	FORGOTTEN	FORGETTING
JEST	JESTS	JESTED	JESTED	JESTING
SEE	SEES	SAW	SEEN	SEEING

READY ?

A total of 40 programs carry out the 102 sessions.

In a number of sessions the student enters the first names of some of his classmates, these becoming, along with his, characters within the lesson:

EUX271

CONNECTORS I

TYPE YOUR FIRST NAME. ? BOB

'CONNECTORS I' HAS 2 LESSONS. WHICH ARE YOU TO DO: 1 OR 2 ? 1

LET'S ACT LIKE I'M A NEW CLASSMATE OF YOURS, BOB.
TELL ME THE FIRST NAMES OF SOME OF OUR OTHER CLASSMATES,
THREE GIRLS AND THREE BOYS:

READY ?

GIRL NUMBER 1 IS ? MARY
IS HER NAME--MARY--SPELLED CORRECTLY (YES, NO) ? YES

GIRL NUMBER 2 IS ? JANE
IS HER NAME--JANE--SPELLED CORRECTLY (YES, NO) ? YES

GIRL NUMBER 3 IS ? SUSAN
IS HER NAME--SUSAN--SPELLED CORRECTLY (YES, NO) ? YES

BOY NUMBER 1 IS ? ROB
IS HIS NAME--ROB--SPELLED CORRECTLY (YES, NO) ? YES

BOY NUMBER 2 IS ? JAMES
IS HIS NAME--JAMES--SPELLED CORRECTLY (YES, NO) ? YES

BOY NUMBER 3 IS ? ROGER
IS HIS NAME--ROGER--SPELLED CORRECTLY (YES, NO) ? YES

BOB, LET'S STUDY SOME SENTENCES:

1. MARY <AND> JANE ARE WHISPERING.

'AND' IS A CONNECTOR.
'AND' CONNECTS 'MARY' AND 'JANE.' 'AND' POINTS TWO WAYS.

READY ?

TWO CLASSMATES ARE WHISPERING! BOTH 'MARY' AND 'JANE.'
BOTH 'MARY' AND 'JANE' ARE SUBJECTS OF 'ARE WHISPERING.'

READY ?

'AND' ADDS 'MARY' AND 'JANE' TO COUNT 2.
'ARE' COUNTS 2 AND AGREES.

READY ?

2. JANE <OR> SUSAN IS TALKING.

'OR' IS A CONNECTOR.
'OR' CONNECTS 'JANE' AND 'SUSAN.' 'OR' POINTS TWO WAYS.

READY ?

There is a significant reason for this, far beyond gaining student interest. As indirectly taught within EUX II, the articles A and THE are far more complex than usually represented in the schoolroom. Rather simplistically, to begin an exercise with THE GIRL IS TALKING is to begin with a non-idiomatic sentence, there being no antecedent. However, assuming that the student accepts the conversational character of the terminal and that he introduces some of his classmates' names, then sentences with identifying prepositional phrases like THE GIRL WITH JANE IS TALKING become idiomatic. Traditional textbooks often ignore the subtlety of such constructions. But stand-alone programs that "generate sentences" handle this challenge reasonably well, not being bound or constrained by static files that parallel the static qualities of textbooks.

Written for the elementary classroom teacher, A CAI APPROACH TO THE DEVELOPMENT OF ENGLISH SKILLS: ENGLISH USAGE EXERCISES II discusses the content, organization, and philosophy of EUX II. It then presents a discussion and outline of each series:

32 POSSESSION EUX311-315

The series 'Possession' composed of eleven sessions, continues a consideration of pronouns (and nouns) introduced in series 13, EUX121-125 ('Possession'), and 14, EUX131-133 ('It' & 'They') of EUX I. This series treats only third person possessive pronouns.

The first session of EUX311 reviews the apostrophe + S form of names and the pronouns substituted for these possessive names:

Jane's friend her friend
Bill's classmate his classmate

Introduced in session one is the connector OF that 'marks' possession:

A friend of Jane's A friend of hers
A classmate of Bill's A classmate of his

This discussion thus presents the third function or use of the connector OF touched upon by EUX II. Practice with these two possessive forms for single names--and their pronoun substitutes--is given in session one. Session two presents double names:

Jane and Bill's friend Their friend
A friend of Jane and Bill's A friend of theirs

The two sessions of EUX312 present practice with noun and pronoun possessive forms before non-people nouns. The possessive OF form does not appear in either session. However, partitive OF forms are extensively displayed in an indirect attempt to keep the student alert to the other 'uses' of OF. In the exercises of both sessions, after an antecedent sentence is presented, the student is asked to choose IS or ARE to suit the sentence counts:

Jane has two pens.
--- one of Jane's pens new?

In session one, the student is then asked to substitute a pronoun:

Is one of > Jane's < pens new?

But in session two, the student is then asked to replace a possessive noun antecedent for the pronoun that appears:

Is one of > her < pens new?

These notes are then followed with displays of each of the sessions of the series.

In summary, the 102 lessons of EUX II extend the developmental and sequential written usage (and grammar) concepts begun by the 96 lessons of EUX I. The teacher's manual gives significant support to the classroom teacher who directs students in the study of EUX lessons. Written in stand-alone form, the BASIC programs of EUX are easily transported. EUX is available to elementary schools for the cost of handling.

1/ 'ENGLISH USAGE EXERCISES I: A CAI Series for the Elementary School,' PROCEEDINGS, Association for Educational Data Systems, Atlanta, 1978.

EXPLORING THE POTENTIAL USE OF COMPUTER ASSISTED INSTRUCTION IN A NURSING CURRICULUM

Patricia J. Morin, R.N., M.S.N.
Julia Ann Fisco, R.N., M.S.N.

In December 1977 and in November 1978, faculty at the University of Nebraska College of Nursing received fully-funded grants of \$1400 and \$2100, respectively, from the University of Nebraska Academic Computing Services. The first grant provided an opportunity to evaluate selected existing computer programs available in medical-surgical and mental health nursing to determine if both the didactic and applied learning experiences for students of nursing could be enhanced. The second grant provided opportunity for computer programs to be evaluated by students to determine if the content is adjunctive to that taught in the curriculum and by faculty for potential use in the curriculum. It also allowed for faculty submitting the grant to explore variables (e.g., preferred learning styles) which influence student use of computer assisted instruction.

Introduction

During the Fall Semester 1977, faculty interested in exploring the use of computer assisted instruction in nursing curriculums wrote and submitted a grant proposal to Academic Computing Services Task Force of the University of Nebraska to obtain funds to evaluate selected, existing computer programs available in medical-surgical and psychiatric-mental health nursing. The major focus of the project was to determine if the existing computer programs provided the opportunity to enhance both the didactic and applied learning experiences for students of nursing. The grant was fully funded in December 1977 and work began on the project in January, 1978.

Programs selected for the grant were determined by the areas of expertise and interest of the participating faculty. The programs were primarily obtained from Ohio State University Division of Computing Services for Medical Education and Research.

Methods Used In Achieving The Goals Of The Project

To facilitate compilation of evaluative comments about the computer programs reviewed, participating faculty developed an evaluation tool. The major emphasis of the tool was to determine whether or not the programs were in keeping with the College of Nursing curriculum (e.g., were the learning strategies used in the program in agreement with those presented in the curriculum?) and could the

programs be adopted for use in a particular course.

Review and evaluation of the programs began in February, 1978. Faculty evaluated specific programs based on their expertise in nursing and interests. A total of seventeen programs were reviewed. Faculty specifically evaluated the programs to determine if the programs used one or more of the following teaching strategies: (1) tutorial methods, (2) testing previously learned content, and (3) problem-simulation. Faculty who reviewed the programs judged that the programs which used the strategy of problem-simulation were most ideally suited for use in computer assisted instruction in nursing curriculums.

Ongoing Project

Faculty judged that some of the programs reviewed could provide viable learning experiences for the students in both the didactic and applied learning experiences. For this reason the two faculty who had reviewed the programs submitted a second grant proposal to the Academic Computing Services in the Fall, 1978. The second grant proposal was fully funded and work began on the project in January, 1979.

This nature of the second computer grant was three-fold: (1) to enable students to evaluate selected computer programs which relate to content taught in the nursing curriculum, (2) to enable faculty to evaluate existing computer programs for potential use in the College of Nursing curriculum, and (3) to enable the

20 two faculty who submitted the grant to explore variables (e.g., preferred learning styles) which influence student use of computer assisted instruction.

An evaluation tool for faculty use which was developed as part of the first grant assisted faculty in evaluating the programs including educational strategies used and appropriateness for the College of Nursing curriculum. An evaluation tool for students was developed for collecting student feedback related to their experience with computer assisted instruction.

Summary

The grants enabled the faculty at the University of Nebraska College of Nursing-Lincoln Division to determine if the use of computer assisted instruction could be an adjunct to the nursing curriculum and also to explore variables which influence student use of the computer.

HUMANISM FOR THE COMPUTER AGE
IN EDUCATION: PLANNING FOR
MAN/MACHINE SYNCHRONICITY

George Donoian, Ed.D.
Business Administrator
Detroit Public Schools

The writer presents six requirements for a humanistic computer-based education. Without them, students would learn mechanistically. Dehumanization could result if program writers, parents, teachers and board members disregard the affects of this process. Hopefully the humanism effort is enhanced when new software includes these essential elements.

INTRODUCTION

The pace of automation in the latter half of the twentieth century has increased so rapidly that in the basic industries workers have become observers in the manufacturing process. Likewise the number of unskilled employees has decreased, replaced by machines controlled by computers. In their place, manufacturing plants have hired highly skilled technicians who keep these machines going under the watchful eye of computers. Additional technical personnel added to modern plants include the computer staff. They too have displaced unskilled laborers who have drastically diminished from the working force. A recent example of this phenomenon is evident at the General Motors Lordstown, Ohio plant. What few unskilled workers there are provide the basic material for the automated machines to stamp out body panels and weld them into unitized automobiles.

Today, hand-dipped chocolate candy is most difficult to obtain for the same reason. Years ago thousands of unskilled, cleanly-clad women produced all the candy. All this was not fast enough or without cost to the candy kitchen owners. Rising demands for union scale wages, fringe benefits, uniform allowances and other requirements caused major confectionary producers to turn to automation in self-defense. Today the story is the same except the women are gone and the surviving factories are filled with efficient machinery run by a handful of skilled operators.

Our purpose is not to lament the passing of unskilled labor from the working world. It is, however, to cite the

changing times requiring technical personnel who keep intricate machinery whirring. These sophisticated operations call for programmed controls provided by the untiring vigil and under the command of computers which are, in turn, run by equally skilled staff. To be sure the computers and the machinery they operate are capital expenditures, justifiable expenses which entrepreneurs are allowed to depreciate within a reasonable time as approved by the Internal Revenue Service. This would not be true of a labor force with its ever-increasing costs and most likely decreasing number of quality units produced. Therefore, it is difficult to see how cost-conscious manufacturers will return to an early twentieth century labor-intensive society when salaries were nominal and employee benefits non-existent.

MODERN TRENDS CONTINUE

Computer controlled machines have other advantages. They do not perform temperamentally; thus, quality and output remain constant. Union slowdowns cannot occur. Machines do not sabotage the works-employees do. As long as the computer guidance and production tools remain operative, the owners can expect so many units per hour to be produced for the marketplace at a prescribed cost.

In contrast, we cannot expect all British automobile manufacturers to return to the Rolls-Royce production schedule of building three hundred cars per week by hand. First, the skills required for building custom automobiles have been

lost long ago and second, thousands of Britons would be thrown out onto the streets if such a decision were made by the mass producers. Therefore, the survival and economic well-being of any viable manufacturer are dependent upon increasing the output of units at a modest cost in order for his product to compete in the retail arena and reap a profit substantial enough for shareholders. Likewise, it is utterly unthinkable for the General Electric Corporation to hire thousands of skilled workers for their electric frypan plant to handcraft every unit and try to sell them at one hundred dollars apiece.

ATTENDANT PROBLEMS OF AUTOMATION

In this headlong drive to automation, industry has often lost touch with workers, their needs, problems and perceptions. For example, the workers' hue and cry at Lordstown was that the work was boring. When was assembly line work exciting, self-gratifying or enlightening and psychologically valuable? The same can be said of any monotonous task anywhere which requires little brain power, some speed and dexterity and lots of resiliency in the face of repetitive effort, hour after hour.

Does the same dilemma portend any significance for education, especially when computers have increasingly taken over the routine teaching tasks of student drill, testing and diagnosis for almost every school subject? In the same thrust for mechanization of the teaching profession, school people have been buffeted by critics who accuse them of unsatisfactory performance as evidenced by poor student mastery, low national achievement test scores and lack of accountability. These charges have been reinforced by newspaper headlines of teacher strikes, contract delays and closing schools with other forms of militancy and demands for increased salaries. In their justification for equity demands in the labor market, teachers have not convinced the public that they are worth their cost. With these forces stirring up the public, curriculum specialists have quietly persuaded boards of education to lease computers. Thus, computer-taught students have learned more basic skills in reading and math in less time, at reduced costs and more efficiently than in traditional classrooms, taught by teachers.

Software development has reached the point in time that the parents of machine age students are convinced by achievement results that their children have learned better sitting in front of

the computer terminals than in traditional classrooms. As more and more school districts turn to computers for instruction and show significant student gains, the overall trend will be a difficult one to overturn or delay when all districts will eventually be tied into an electronic box for instructional as well as for administrative purposes.

What caution must school boards, parents, teachers and students observe? When such a critical mass as the majority of districts are utilizing computers for educating youngsters? Who will be serving as their responsible adult role models? Who will be providing the humanistic training for computer-educated youngsters and filling their affective need when the school day is preoccupied with cognitive skills building? Just how much warmth and personal recognition can a computer generate? Where will the teachers be when the computers preempt the students' total time in school? Could the students then judge their schools to be boring? Who but the key-punch operators can sit for hours before a computer console and video display tube without some loss of interest? Since so much of computer-based instruction requires repetitive and prescribed response on the part of the students, won't they soon realize that efficiency and mastery have their inherent problems too? Will they not come to the conclusion that efficiency has its own cost to extract from them?

IMPORTANT EDUCATIONAL ISSUES AT STAKE

Soon researchers will be issuing the results of a study attesting to the validity of the questions advanced here. However, before that happens, this writer hopes that software developers, teachers and parents will heed the call for planned humanism education in our schools. Nothing contrived or artificial will work, only genuine human concern and interaction which machines cannot provide or guarantee, must be an integral part of the total curriculum of every school and experience of every student.

A real danger is to lose sight of the art of being human as schools become more factory-like in producing automated learning units, not necessarily students. Only a balance can check the achieving pupils from becoming too engrossed with the science of learning at the expense of their underdeveloped humanity. Already computer publications contain articles written by concerned people describing the trend of dehumanization manifesting itself in the software in our schools. First they speak of safeguarding the confidentiality of data stored in computer banks and how compro-

mising methods could destroy a system's security. More crucial is the matter of students becoming mere numbers while the data take on an aura of infallibility. The trend is well nigh complete they say. Witness the computer's knowledge of your income tax situation, charge accounts and driving record.

When complete responsibility for teaching is turned over to computers because they are effective managers, fundamental human contact and interaction will become nearly nonexistent. What little humanity we have in schools will be further reduced. Therefore, we should seriously consider the effects of unheeded warnings and unbridled consequences of a computer-managed education for our children. Overemphasizing subject matter mastery at the expense of damaging what humane qualities students possess and worse yet, destroying any growth potential instead of enhancing it, should be a great concern to us who believe that education makes a person humane. Otherwise, what learning students will gain will be only training. Certainly no enlightenment without an education can be expected to be programmed into the software at this time. How can we then expect a Renaissance-type being to be produced by the computers, who thinks for himself, is capable and confident, equally at home with physical and social sciences? This is an improbable paradox, to be sure.

Avoiding these issues or minimizing their long-term effects would be foolhardy. To produce a society of trained beings with little discernable affect would be tragic. The utter consequences of such a society in power is frightening. Present governmental leaders serve by the consent of the public; most of them are not always logical, precise or right. They make mistakes in judgment, calculate incorrectly and perceive circumstances differently. All these traits indicate a certain amount of humanity, adjustment and potential for learning from others. Where will future leaders gain these human qualities if they are the products of a controlled training program which excludes humanistic values and skills?

HOW WE CAN AVOID DEHUMANIZING STUDENTS

This paper is presented not to inhibit or delay computer technology and software development. On the contrary, it is my intent to encourage growth in these areas with a definitive bias. It is hoped that schools do not replace humanism with achievement mastery and idolize hardware and software over the value of the individual student. We should never be guilty of extolling the virtues of machine-age education at the

expense of any child. Just to hope that software developers are going to include humanism programs would be naive of those of us who are responsible for making curriculum decisions. Furthermore, just to have software developers assure us would be equally naive. We need more than their word; we need to know how they will devise computer-based education with humanistic values and skills. It is not that developers are incapable, but we question their motives for not including these vital elements. It is far easier to exclude the affective domain serving the humanistic movement than to take a great deal of time, money and effort to include them.

Education should prepare students to be compassionate toward others, for self-understanding and to value the importance of others. These objectives are more important than remembering merely that King John signed the Magna Carta in 1215 A.D. giving the English people basic rights, for example. Likewise no computer is going to convince any student that it is a warm, empathic being who cares by displaying the message on the video display: "Good morning, Jim. I expect you to do as well today as you did yesterday with 10/10 correct". Granted this much is only a little better than nothing, but hardly any measure of humanity which caring, helping teachers give daily to students.

PLANNED MAN/MACHINE SYNCHRONICITY

I believe that humanism and software programming will not naturally converge to produce the outcomes we hope for. Computer curriculum writers and educators must realize that for humanism to become a program of study, they must take deliberate action to include the skills and values for students to learn. Definitive plans must be made by all who are responsible for education that nothing is left to chance.

In this respect, I am advocating a planned synchronicity, that is, a direct relationship be developed between the objectives of humanistic education and the software. I do not believe as Dr. Carl Jung, the late Swiss psychiatrist, that synchronicity is a "concept diametrically opposed to that of causality.....the coincidence of events in space and time as something more than mere chance, namely a peculiar interdependence of objective events".....

—¹Ira Progoff, Jung, Synchronicity and Human Testing: Noncausal Dimensions In Human Experience (New York: Dell Publishing Company, 1973), p.23.

I would redefine synchronicity to mean planned outcomes so that we can expect results as we set them up to happen, particularly as we apply the concept to schools. Too often we as parents and teachers hope for much more than we witness in learning outcomes because of our inadequate planning and teaching. With the computer to aid us in managing instruction, we should expect subject matter to be better taught and learned. This is especially true for affective domain areas like the humanities. Therefore, parents and board members alike will be assured that the teaching and practice of humanism can be both planned and spontaneous, genuine, warm and encouraging while helping and developing youngsters.

TOWARD A HUMANISTIC CURRICULUM

Not all contemporary schools are bastions of humanistic practice. Granted, there are some teachers whose classroom din resembles the negotiation sessions between management and labor before the serious bargaining begins, particularly punctuated by exchanges of threats, obscenities and personal abuse. However, most teachers are friendly, helpful and instructive persons who fulfill their mission by teaching in their specialized field, serve as model adults for their students and represent the core values of the prevailing times and culture. It is rare to find the former kind. In most cases, teachers are like the latter characterization who understand, encourage and practice humanism day in and day out, without regret or reward. Therein lies the problem: how will we as teachers be able to guarantee the development and practice of humanism when the curriculum becomes more and more mechanized? How can computer-based instruction include basic values, societal mores, humanistic traditions and development of self through sharing with others? I believe we are in imminent danger if we do not include the following six requirements for a humanistic approach to computer programs.

Active versus Passive Response

One safeguard which will serve the centrality of the human being and its self-development that must be incorporated into electronic lessons is the active participation of the individual in responding to learning queries. When only his passive, automatic selection of an appropriate response is required, taken from a choice of four, for example, then the student is relegated to a passive mode. Then he merely reads, studies some graphics, listens to a canned presentation and responds correctly. We would assume that

he has learned something. After studying ten lessons and answering questions correctly, he will be diagnostically declared a "master" of the unit.

Maybe so, but what price has he paid for this mastery? He has lost any semblance of skills in exploration, random discovery, self-directedness, analysis and synthesis, or using any critical or logical thinking processes to arrive at an answer. The machine has provided the guidance for him because the program may call for a ninety percent mastery of the material. So our well-behaved, achieving students become conditioned automats of the programmed curriculum.

Personal versus Machine Purposes

Close akin to the first humanistic imperative is the preservation of the learner's purposes. We know that a student learns most because he sees value in the knowledge to be gained and skills to be mastered. He might go as far as to adopt the teacher's stated purpose for learning. However, with computer-based software aimed at all students and presuming that all students perceive the same purposes, the student's purposes do not even merit consideration. Learning in the electronic age is external to the students. What may be important motives for learning to the learners are no longer factors influencing their pursuit of knowledge and skills. Program objectives provide the incentives for achievement, while goal mastery is the prime motive. The pursuit of knowledge for its own sake is null and void.

Variability versus Infallibility

Imagine the sterility of computer-based education characterized by a lack of lively debate, argumentation and protestation. The packaged curriculum circumscribes the learning parameters and limits them to queries by the machine that determines the success of the student. Therefore, there is no room for argumentation or debate. The word "issue" cannot exist in a closed structure where mastery is predictable when eighty five percent of the students will respond correctly. Thus the computer program dictates the game and the rules, while the students respond accordingly or lose. In any case the loss is graver than anyone can imagine. Where will students learn the fine art of logical questioning and to develop forceful, convincing arguments? How will they analyze data, fact from fancy and hearsay quickly while they compose speeches, themes and answers to essay questions—all necessary skills for learning to engage in when studying issues of every persuasion?

All the computer program provides is an unimpassioned set of questions with its inherent logic that cannot be conveyed to the students by magic or osmosis. For students to gain special skills there is no substitute for real-life experiences with moot courts, student trials and debate exercises which provide the substance for contrasting viewpoints and emotional persuasions. No machine program can be generated which can enliven a classroom full of students to a fever pitch.

Time Management versus Pacing

Close akin to machine purposes is the computer's management of time. It is a presumption made by the program developer that a particular subject is to be completed in a prescribed time, based on predictable achievement parameters of the so called "average" student. No one can set his own input rate for either the video and/or audio portion of the teaching stimulus material. Only when great numbers of students fail will there be any attempt to modify the computer program for a different pace. Then the developer will reset the exposure rate to what he determines to be comfortable. Thus students are not able to set their own pace for learning. Program writers who believe that time management cannot be left up to learners to decide and regulate, further depersonalize the classroom. This time the learning pace forces students to keep up or slow down to an average rate calibrated on some mythical mean group of students who are neither too bright nor too dull. The average could easily become the standard of excellence for the entire class, controlling the educational destiny for the next generation.

Personal versus Computer Wisdom

No computer program will be developed which will encourage, promote and reward student excursions into the world of self-directed learning. How could students so inclined take such trips into wide-ranging topics seeking out trivial, obscure and useless bits of information? This explorative trait would contribute to students' growth, enhance their unique personalities and unusual affects. I cannot foresee a program writer wasting his time playing cerebral leapfrog among his billions of information synapses to float freely down a stream of consciousness, concocting fanciful lessons in diverse fields for a leisurely course of study. These pursuits of happiness are too time-consuming, costly and ineffective ways to learn. The rewards for such mental meanderings are too individualistic and most of all, they cannot be charted

on student achievement profiles because they do not lend themselves to discreet and convenient packages of mastery skills. In no way will children be best served by capsule learning units presented for an inquiring mind whose spirit may become imprisoned by a black box - all without challenge, excitement or romance in which discovery of trivia, the obscure and useless knowledge are a big part of learning.

Sharing versus Machine Care

If all else were sacrificed for the cause of achievement via computers, the one which cannot be, is sharing with others what one has learned. This significant force is the keystone of humanism where we must draw the final line. Eliminating peer interaction, teacher-to-pupil exchanges and robust class discussion for machine care is totally inhuman. There can be no love built into computer programs that provide students the thrill of winning a debate or being chosen as valedictorian. What can a video display terminal do for students who score perfectly on an essay question? There is no suitable substitute for the ego-building experiences students feel when their success is acknowledged by their classmates and teachers. Learners require a nurturing environment populated by an appreciative and affirming audience.

WHAT SHOULD WE DO NOW?

We as teachers should lead the march on incorporating the core humanistic values into the software destined for our students. This is why I advocate a planned synchronicity - a guaranteed inclusion of the basic requirements for a curriculum intended for an educated society. It is my hope that humanism is the key element in our students' schooling. To accomplish this task, I propose the following minimal expectations permeate the whole curriculum as computer technology advances to the head of the class and assumes the burden of teaching:

1. Students must be actively engaged in pursuit of knowledge, skills and appreciation of humanistic values throughout their school careers. They must not be trained to be passive receptacles of programmed materials.
2. Humanism cannot be advanced without an accounting of the students' purposes for learning. Computer logic and streamlined software cannot substitute for the individual's needs and wants. Machine purpose can only be external and demanding, never personally rewarding and intrinsic.

3. Machine infallibility can only defeat students, rendering them impotent, whereas human variability provides each student with individual traits and insights - unique features which makes them different and therefore, excitingly unpredictable. Encourage it.
4. When machine programming is required for our students to direct when they should learn what, then we as human beings are no longer superior and the humanity in us has diminished or worse yet, disappeared if we must forego our judgment as to how we should conduct our education.
5. The pursuit of happiness is a sure sign of being human, but if we forsake our quest for trivia, the obscure or what might be deemed useless information, then for certain we have surrendered our humanity to the computer program, the content of which must then be considered sacred.
6. Sharing our learning experiences with others is the ultimate fun of schooling and humanistic endeavors. No video tube or computer program can ever provide the audience teachers, peers or parents are for students.

We insist that program developers include planned synchronicity in their software. It must include and enhance humanistic skills and knowledge as described or we should not permit them to peddle the wares in our schools. What I fear most is the unquestioning acceptance of the present state of the art in software, to be comfortable with it. What is available now is too inadequate for man/machine synchronicity to guarantee any humanistic qualities in our schools. I further believe that there are far too many dangers in leaving educational programming to writers who would exclude humanistic values from their work.

Ultimately we must educate ourselves in the sciences and arts, but this can only come about by careful planning and skillful attainment of valued objectives centering around the importance of people, not machines, however sophisticated they may become.

MAGNET MATH - DRAWING THE BEST FROM YOUR STUDENTS BY COMPUTER

Betty Vanden Bosch

Staff Assistant, Mathematics

Flint Public Schools

The Magnet Math Program, one component of a large and varied base of computer resources for the Kindergarten to Twelve curriculum in the Flint Schools, is presently under development by Mrs. Miriam Schaefer and me. This program is designed to provide not only a complete arithmetic package of drill and practice in specific skills for use by the classroom teacher, but CAI to support our present secondary courses from Pre-Algebra through Trigonometry. Among its attributes are its broad scope, flexibility and the organization of its material, designed especially for easy use and record keeping in the classroom. This approach to drill and practice should be of interest to any school system considering locally written materials.

Since 1968, the Flint Community Schools have instituted a number computer based programs for their kindergarten to twelfth grade program, using three Hewlett Packard computers. Among these programs are:

Computer Assisted Instruction for the Title One program at the elementary level

A high school course called Computer Math for advanced mathematics students, aimed heavily at computer programming involving mathematics topics

A series of workshops designed to promote computer literacy among administrators, teachers and the community

A course designed for the adult high school

A specialty at the high school level in data processing

Both CAI and administrative use in high school science classes

The Magnet Math Program

What the Magnet Math Program is and What It Provides

Certain schools in the Flint School System are designated as Magnet Schools, providing specialty courses in such areas as mathematics and science, humanities and fine arts, business data processing,

and many others. The name of the Magnet Math Program was taken from this concept and has been designed for, but not limited to, the schools designated as Math-Science Magnet Schools.

The three major parts of this program are:

- (1) a list of code numbers and descriptions of the various lessons that can be found in the program,
- (2) the computer program which accesses these lessons, and
- (3) the report to the teacher, giving student names, scores, and time needed for each lesson.

The Magnet Math Program, under development by the author of this article and Mrs. Miriam Schaefer, Coordinator of Mathematics for the Flint Schools, is designed to provide not only a complete arithmetic package of drill and practice in specific skills for use by the classroom teacher, but computer assisted instruction to support the present secondary courses from Pre-Algebra through Trigonometry. Among the attributes of the Magnet Math Program are its broad scope, flexibility and the organization of its material, designed especially for easy use and record keeping in the classroom. In making this presentation, it is hoped to promote interest of other school districts in building similar programs aimed at:

- 28 (1) developing some computer literacy through actual usage of computer equipment.
- (2) providing a large amount of drill and practice with immediate feedback concerning errors and, if necessary, correct responses. This not only reduces teacher time spent in search for computational errors but puts extra responsibility on the student to combat his own errors (good for self esteem!) and leaves more teacher time free for dealing more with errors in method as opposed to errors in computation.
- (3) providing in many components of the program a changing supply of drill material with immediate scoring for the student and record keeping already done for the teacher. (No two students are expected to be presented with exactly the same lessons.)
- (4) providing a logical sequence of lessons which can be used at the secondary level, matching the order of textbooks in current use, coded to be easily identifiable to the teacher as to lesson content and time of use.
- (5) providing flexibility in that the material may be used when appropriate and omitted when not (very few lessons are given in blocks--most are independent of one another) and in that these program components at the secondary level can be re-organized to assume the sequence of new textbooks when adopted.
- (6) providing arithmetic components broken down into specific and identifiable skills which can be used at any grade level where appropriate without problems ordinarily encountered with graded materials.
- (7) providing diagnostic tests among the arithmetic components to determine where the student needs specific help.
- (8) avoiding the confusion encountered when looking at a long list of computer programs available, not knowing which are appropriate in content and for a given student level.

How the Magnet Math Arithmetic Components are Used

In one of the elementary schools designated as a Magnet School, a teacher may have one, some or all students take an inventory test on whole numbers (this would be one of the components of the

Magnet Math Program.) After deciding which components in that area are appropriate for a specific student, the code numbers may be noted on an index card.

The student takes the card with code numbers to the computer terminal with him and at the proper cue enters the code number. When the student makes an error, he is given a second chance to respond, and if he is successful the second time, the computer responds with the next problem. However, should he miss it twice, the computer responds with the correct answer before going on to the next problem.

When the lesson is completed, the student receives a message telling him how many problems he had correct on the first try and how many he had correct on the second try. This information is then recorded for the teacher, along with the date and the length of time required to do the lesson. Some lessons even finish with a list of the student's mistakes and the correct answers so that he can study what he has missed. Since hard copy is provided, it is conceivable that the student may take his printout home and get help in a weak area, e.g. multiplication facts up to 5×5 .

How the Magnet Math Secondary Components are Used

In a junior high school designated as a Magnet School or in any of the four comprehensive high schools, students presently in Pre-Algebra, First Year Algebra, or Geometry, may be assigned singly, or as a class, to do a specific lesson by computer. It is suggested to teachers using this form of computer assisted instruction that the code numbers to be used be placed on an index card as is done with the arithmetic components for elementary students.

The student enters the number code requested by the computer and receives a drill and practice lesson or a vocabulary lesson. The lesson is presented to the student, allowing a second chance in case of error, then responding with the correct answer, should the item be missed twice. Vocabulary lessons are drawn from a bank of words and their definitions; computational lessons are created randomly to prescription. While definitions are taken from a bank and selection is therefore limited, lessons of computational problems may be used a number of times without problems being repeated.

Comments, et cetera

The advantages of using this type of computer assisted instruction include:

1. The problems are, in most cases, randomly generated, so that no two students are presented the same set of problems. This also means a student can repeat the same lesson, but with a different set of problems.
2. There is immediate feedback on errors with a second chance allowed to correct an error.
3. At the end of the day, the teacher receives a report, similar to the one attached of all students who have worked at the computer since the last time the report was requested. This means that the lesson has been generated, scored and recorded, giving the teacher opportunity to spend time with students in other ways.
4. The students respond most favorably to working with computer equipment and would spend more time at the terminals, if the time were available.
5. Teachers do not have to search long lists of program titles to find suitable materials, since CODES is a list organized for the classroom teacher. CODE 8241 communicates to the teacher that this is the first computerized lesson (1) of chapter four (4). The entire 8200 series is written as backup for our present Algebra I course.
6. Arithmetic components are broken down into specific skills so that a student can work intensively in a specific area of weakness.
7. The teacher controls the lessons used and may individualize as much as is desirable, since a lesson may be requested at any time.

No statistical results are available at this time to indicate that this is a more successful route to take than traditional teaching. Three reasons that this has not been done are:

- (1) the Magnet Math Program is not complete
- (2) the Magnet School concept is still in its infancy and not ready for statistical evaluation of this kind, and
- (3) the Magnet Math Program is used only by teachers who choose voluntarily to use it and then when they choose to use it

Enthusiastic reports and consistent usage indicate that the program is being used

as a valuable adjunct to more traditional classroom teaching.

Below is a partial list of the nearly six hundred lessons available in the Magnet Math Program.

FLINT COMMUNITY SCHOOLS MAGNET MATHEMATICS PROGRAM

ASSIGNMENT CODE LIST PAGE 1

CODE	ASSIGNMENT DESCRIPTION
8211	DECIMAL ADDITION
8212	DECIMAL SUBTRACTION
8213	DECIMAL MULTIPLICATION
8214	SET VOCABULARY
8221	FRACTION ADDITION
8222	FRACTION MULTIPLICATION
8223	FRACTION DIVISION
8224	EQUATION VOCABULARY
8225	*GEOMETRIC TERMS
8226	WRITING EQUATIONS
8231	PROPERTIES, VOCAB.
8232	SIGNED NUMBER DRILL
8233	COMPUTING ABS. VALUE
8241	ADD REAL NOS. TO EQUATIONS
8242	DIVIDE EQUATIONS BY REAL NOS.
8243	COMBINATION OF CODES 8241, 8242
8251	VOC. OF SETS, INEQ. AND ABS. VAL.
8252	UNDERSTANDING SETS
8253	VENN DIAGRAM I
8254	VENN DIAGRAM II
8255	SET UP EQUATIONS/ABS. VAL.
8256	SOLVING EQUATIONS/ABS. VAL.
8261	VOCABULARY - POLYNOMIALS
8262	POLYNOMIAL ADDITION
8263	POLYNOMIAL MULTIPLICATION
8271	PRIME NO. INSTRUCTION
8272	DRILL - PRIMES
8273	COMPETITION - PRIMES
8274	PRIME FACTORS

FLINT COMMUNITY SCHOOLS MAGNET MATHEMATICS PROGRAM

WHAT IS YOUR NAME? BETTY VANDEN BOSCH
 WHAT ASSIGNMENT CODE DO YOU WANT? 8241

ADD REAL NOS. TO EQUATIONS 3/14/78 4:21 PM

AN EQUATION CAN BE TRANSFORMED BY ADDING THE SAME REAL
 NUMBER TO BOTH MEMBERS OF AN EQUATION OR BY DIVIDING BOTH
 MEMBERS BY THE SAME REAL NUMBER.
 TELL WHAT REAL NUMBER SHOULD BE ADDED TO EACH MEMBER OF
 THE FOLLOWING EQUATIONS.

STUDENT RESPONSES
 ARE CIRCLED

ADD WHAT NUMBER? -4

$$-7X + 4 = 21$$

$$-4 \quad -4$$

THIS TRANSFORMATION PRODUCES: $-7X = 17$

ADD WHAT NUMBER? 2

$$3X - 2 = 31$$

$$+2 \quad +2$$

THIS TRANSFORMATION PRODUCES: $3X = 33$

ADD WHAT NUMBER? -10

$$3X + 10 = 38$$

$$-10 \quad -10$$

THIS TRANSFORMATION PRODUCES: $3X = 28$

ADD WHAT NUMBER? 4

$$-9X - 4 = 27$$

$$+4 \quad +4$$

THIS TRANSFORMATION PRODUCES: $-9X = 29$

TRY THAT ONE AGAIN

THIS TRANSFORMATION PRODUCES: $-9X = 31$

RIGHT ON
 SECOND TRY

YOU HAD 7 PROBLEMS CORRECT ON THE FIRST TRIAL.
 THAT IS A SCORE OF 70%.
 ALSO, YOU HAD 2 MORE PROBLEMS CORRECT ON THE SECOND TRIAL.

SESSION TIME: 3 MINUTES. 3/14/78 4:24 PM

FLINT COMMUNITY SCHOOLS MAGNET MATHEMATICS PROGRAM

WHAT IS YOUR NAME? HI THERE

WHAT ASSIGNMENT CODE DO YOU WANT? 420

WN/ DD/D=D R D, FACTORS < 6

3/1/79 3:35 PM

TYPE THE ANSWERS THAT HAVE A REMAINDER AS 5 R 2.

- (1) $3 \overline{) 12}$ 4
- (2) $1 \overline{) 1}$ 1
- (3) $5 \overline{) 27}$ 5 R 2
- (4) $4 \overline{) 18}$ 4 R 2

STUDENT RESPONSES
ARE CIRCLED

NO. LET'S TRY THIS.

- $2 \times 4 = 8$
- $3 \times 4 = 12$
- $4 \times 4 = 16$
- $5 \times 4 = 20$

RIGHT ON
SECOND TRY

NOW TRY AGAIN.

- $4 \overline{) 18}$ 4 R 2
- (5) $2 \overline{) 7}$ 3 R 1
- (6) $2 \overline{) 11}$ 5 R 1
- (7) $3 \overline{) 15}$ 5
- (8) $5 \overline{) 13}$ 2 R 3

- (13) $4 \overline{) 10}$ 2 R 2
- (14) $2 \overline{) 5}$ 2 R 1
- (15) $3 \overline{) 11}$ 3 R 2

YOU HAD 14 PROBLEMS CORRECT ON THE FIRST TRIAL.
THAT IS A SCORE OF 93%.

ALSO, YOU HAD 1 MORE PROBLEM CORRECT ON THE SECOND TRIAL.

SESSION TIME: 2 MINUTES.

3/1/79 3:37 PM

32 An example of the daily terminal log kept for the teacher follows.

REPORT

DAILY TERMINAL LOG 3/15/78 9:10 AM

NAME	ASSIGNMENT	RESULT	TIME
BETTY VANDEN BOSCH	ADD REAL NOS. TO EQUATIONS	70%	3
BETTY VANDEN BOSCH	DIVIDE EQUATIONS BY REAL NOS.	80%	2
BETTY VANDEN BOSCH	COMBINATION OF CODES 8241,8242	80%	4

DONE

MICRO IN A MACRO WORLD:
CURRICULUM DEVELOPMENT
FOR THE MICROCOMPUTER

Shelley Yorke Rose
Dallas Independent School District

The Instructional Technology department of the Dallas Independent School District has undertaken to adapt concepts from the district's existing Baseline curriculum for microcomputer CAI use. Under the guidance of Dr. Tom Krueck, elementary level math and reading strands are being developed for subsequent programming and school laboratory use. Peripheral hardware also being tested for possible inclusion are video discs and computer controlled audio. The major limiting factor encountered thus far is the microcomputer's restricted memory. The curriculum writers' two-fold task then is to develop unique CAI curriculum appropriately adapted to the characteristics of both the microcomputers and learner population.

INTRODUCTION

The Instructional Technology branch of the Dallas Independent School District department of Research and Evaluation was created during the 1977-78 academic year. In this, its second year of operation, we have witnessed the de-bugging and marketing of a K-8 computer assisted instruction mathematics curriculum and the inception of a computer assisted instruction remedial reading program. Both curricula have been devised using the existing Dallas Independent School District Baseline, which is an outline of K-12 mathematics and communications skills. Extrapolation of behavioral objectives from the Baseline has served to ease the transition from classroom to computer assisted instruction (CAI).

Devising CAI curriculum according to specific district needs is a relatively new concept in curriculum design. Heretofore, many school districts have purchased standardized CAI programs from such companies as Computer Curriculum Corporation. As a rule, students are then placed at the beginning of an appropriate set of modules and progress sequentially through the curriculum. The Dallas Independent School District Research and Evaluation department has, however, taken a slightly different tack. The student population has been evaluated and directives for CAI design have evolved directly from those areas in which Dallas students need greater focus and aid.

The development of a CAI remedial reading

curriculum grew from the Dallas Independent School District's research into and diagnosis of student needs. Data collected last year revealed a significant decrease in the number of students reading on-level between the third and fourth through sixth grades. Whereas approximately 34% of Dallas Independent School District students were reading on-level in grade 3, only 15-25% of students in grades 4-6 maintained their on-level placement.¹

These percentages of Dallas Independent School District students reading below level correlated with Title I's prerequisite percentages for remediation, expressed as a sliding scale based on grade level. The discovery of the pressing need for student remediation coupled with the availability of funds to purchase the hardware integral to implementation of that remediation through Title I, made a strong case for the development of a CAI remedial reading program for grades 3-5. The medium selected logically complements the existing CAI mathematics program and will serve as a foundation for future K-12 reading modules to be developed upon completion of the remedial reading modules next year.

TASK

The curriculum writers' task objective was developed by Dr. Tom Krueck (Principal Planner, Instructional Technology) after conferring with Dr. William Denton (Deputy Assistant Superintendent, Curriculum Research). This objective is to use CAI

34 to: a) bring below-level students up to level and b) provide drill and practice in deficient skill areas. The target population for the program consists of those Dallas Independent School District students in grades 3-5 who are not classified as either "coping" or "bilingual" but who are reading at approximately 1-2 grade levels below their peers.

METHOD

Before proceeding with actual curriculum development, the curriculum writing group familiarized itself with the Dallas Independent School District Baseline and its auxiliary components. Structured as an informal hierarchy, the Baseline proved useful to the building of hierarchies for reading skills for levels 1-3 to be used for remediation of students in grades 3-5. To support these hierarchies, we further acquainted ourselves with each of the five state-adopted texts: Ginn, Harcourt Brace Jovanovich, Harper & Row, Holt, Rinehart and Winston, and Houghton Mifflin. The methods of instruction presented in these texts helped to both verify and clarify the skills listed in the Baseline.

Our synthesis of this background material resulted in the formation of four reading strands: 1) Phonics, 2) Structural Analysis, 3) Vocabulary, and 4) Comprehension. Phonics and Structural Analysis had been grouped together in the Baseline under the heading of "Word Attack Skills". It was decided to distinguish between the two in deference to textbook presentation methods and writing convenience.

Working definitions have been given to each strand. Phonics involves the identification and reading of letters and vowel and consonant sounds. Structural Analysis is concerned with the reading and use of prefixes, suffixes, syllabication, compound words and contractions. Vocabulary skills are those dealing with the reading of basal sight vocabulary, context clues for word meaning, synonyms and antonyms, and homonyms and homographs. These three strands lead logically to the Comprehension strand, where the concepts learned in the previous strands combine to enable reading for detail, sequence, main idea, and outcome prediction.²

From these strands, we were able to build learning hierarchies for each level and define behavioral objectives for each skill. Research, common sense, and consultation with the reading experts who devised the Baseline led to the creation of hierarchies whose progressions are logical but often arbitrary. Reading research indicates that there is no definitive order inherent in the teaching of certain reading skills.³

Skills have been grouped by the Baseline levels in which their classroom instruction to mastery occur. We similarly employed the grouping of skills by level to insure appropriate student placement for remediation in addition to encouraging teacher use of the program. The emphasis, it should be noted, is to be placed on the skill and not on the level in which the skill is situated since these divisions were made primarily for ease in design and student placement.

PROJECT DESIGN

The curriculum writers, after having devised the four reading strands, selected the most suitable strand on which to work, using expertise and interest as criteria. Module planning guides were then designed to enhance design uniformity and enable the programmers to simultaneously scan module items and item design while programming.

When all modules for one level have been completed, they will be given to the department's programmers, with whom the writers intend to work closely to ensure a smooth transition from design to programming to student use.

The unique nature of the reading program is such that its media recommendations may not necessarily coincide with the current capabilities of available hardware. This is a period where technological development is occurring so rapidly that it is difficult to designate specific hardware for program implementation until most of the proposed curricula is already written. Although media needs will differ between levels and strands, a recurring list of media needs should emerge from a comprehensive evaluation of the written modules. Two limiting factors already encountered relate to microcomputer characters and graphics. Alphabet instruction produced thus far requires the use of both upper- and lower-case letters, while certain vocabulary modules require the use of recognizably representational graphics. It is hoped that other media common denominators will be discerned as the curriculum writing progresses since their discovery will aid in media selection.

Various media alternatives are being considered to meet module requirements. A simple interface with a slide projector could be used to supplement computer graphics; however, this technique would complicate the randomization of item presentation. In order to achieve a semblance of upper- and lower-case letters on a microcomputer equipped with only upper-case characters, graphics could be used, although the memory required for such an endeavor might be prohibitive.

While the future utilization of floppy discs and video discs is being considered to enlarge currently available microcomputer memory, budgetary considerations and hardware availability now serve to define memory as that which is presently marketed.

The utilization of various voice synthesizers is also being considered for reading-readiness modules, supplemental reading activities, and giving lesson instructions. Further investigation into available products will be conducted before specifying audio hardware. Students currently enrolled in the Dallas Independent School District's Bilingual CAI program do seem to comprehend the synthesized speech despite its noticeably Scandinavian inflection. Emphasis must be placed on programming attempts to minimize the projection of accented speech if it is to be used to teach phonics or word identification. The use of cassette tapes for spoken lessons and/or instruction is also being considered as a possible alternative.

After presenting these media limitations, it could be asked why microcomputers have been selected when minicomputers might deal more easily with these problems. The answer can, quite simply, be found in the cost difference between the two. In the pamphlet accompanying the Dallas Independent School District Microcomputer Mathematics Program, K-8, it is stated that:

The introduction of the mini-computer encouraged more school districts to experiment with computer assisted instruction, but still the cost exceeded \$100 per student per year (based on a five year amortization schedule). The introduction of the microcomputer brings the cost of computer assisted instruction within the cost range of every school district and every classroom. Using an equipment cost of \$1,000 for a microcomputer teaching station, amortizing the costs over five years, allowing ten minutes per student per day, and using the teaching station five hours per day, the equipment costs per student per year would be \$6.67; even including software, the cost per student per year could still be \$10.00 or less.⁴

Another point in the microcomputer's favor is its accessibility and operational ease. The Microcomputer Mathematics Program pamphlet goes on to say that:

The microcomputer has basically

the same instructional capabilities as the larger computer; it is very easy to operate, does not require a trained technician, is very portable and may be used in a regular classroom. A training session of one hour or less will prepare teachers and students to operate the microcomputer for CAI purposes. The simplicity of operation and the reliability of the microcomputers make them ideal for use in the classroom and in learning laboratory settings.⁵

PROJECTED PROGRAM OUTCOME

It is expected that remedial reading CAI modules for grades 3-5 will have been designed, programmed, field-tested, and revised by August, 1979. A complementary computer managed instruction system is being designed simultaneously to provide criterion referenced testing for student placement and progression. Teacher manuals and student progress charts will accompany the program into Dallas classrooms in an attempt to provide an accessible yet comprehensive system of instruction.

What began as a simply conceived and stated task has grown into a multifaceted systems approach to instruction and learning. The development of this remedial reading curriculum now depends not only on the curriculum writers but on imaginative resource people, programmers, and teachers as well. The success of the program, however, will not so much lie in the designers' and implementers' satisfaction, but rather in the expected achievement of the student users. Much as the developers desire to claim educational triumph and validity for the microcomputer, it is ultimately the students' triumph which must be sought. It is hoped that proper use of this powerful modern tool will enable the student to experience success in areas of previous failure, thus paving the way for greater learning and achievement.

FOOTNOTES

¹Paul Snelus, telephone conversation, (Dallas Independent School District, Research and Evaluation, Feb. 28, 1979).

²Communications K-12 Baseline, (Dallas Independent School District, 1978), pp. 12-15, 22-26, 32.

³Dr. Al Granowsky, Frances Middleton, Janice Montford, group discussion, (Dallas Independent School District, Jan. 9, 1979).

36⁴ "Microcomputer Mathematics Program, K-8",
(Dallas Independent School District,
1979), p. 1.

⁵ Ibid.

REFERENCES

Communications K-12 Baseline. Dallas
Independent School District, 1978.

Cook, M. Staff training seminar, Region
10, Dallas, Texas: Feb. 8, 1979.

Davies, I. Staff training seminar,
Region 10, Dallas, Texas: Feb. 12,
1979.

Dick, W. and Carey, L. The Systematic
Design of Instruction. Glenview,
Illinois: Scott, Foresman and Co.,
1978.

Eisner, E. Staff training seminar,
Region 10, Dallas, Texas: Jan. 11,
1979.

"Microcomputer Mathematics Program, K-8",
Dallas Independent School District,
1979.

Snelus, P. Telephone conversation, Dallas
Independent School District, Feb. 28,
1979.

"Université du Québec"

PLATO system

by Paul-André Desjardins
Ecole nationale d'administration publique

This paper presents some of the features of the "Université du Québec" PLATO system. "Université du Québec" is a rather special university, french speaking and spread over a land more than twice the size of Texas. Translation and telecommunications problems (and solutions) are reviewed, original french courseware is emphasized.

The university itself.

"Université du Québec" is a young one, now celebrating its tenth anniversary. It has four complete campuses in the cities of Chicoutimi, Montréal, Rimouski and Trois-Rivières, plus half a dozen of specialized institutions in Hull, Montréal, Québec and Rouyn.

Rightfully enough, it has been called the 'university of telecommunications', pioneering such tools as bi-directional audio-video links between classrooms in different cities.

It is now experimenting the PLATO system with a CDC Cyber 73 computer located at head office in Québec city. Service is provided to some 59 terminals, 33 inside and 26 outside "Université du Québec".

Language.

The PLATO system, as born at the University of Illinois, speaks english. Quite natural! It has english documentation, english courseware, english consultants, etc. However "Université du Québec" speaks french and needs its own documentation (see Bibliography) and expertise. Authors have to be comfortable with the system: close to a hundred Aids lessons (the TUTOR language online documentation) have been translated. Most of the good PLATO lessons about PLATO have been translated too.

But we have to get further: french authors have something to teach to french students. They have to choose whether they create something new (see 'French courseware' below) or whether they find a good university-level lesson and translate or adapt it. Translating has been possible for FORTRAN lessons, for example, but most of future work might be adapting because of all the cultural values that go into a lesson, believe it or not. An apparently harmless

question like "Who is on the dollar bill?" has to be adapted. The answer "George Washington" can't be easily found unless the student spends much of his time taking vacations in Atlantic City or Miami. Moreover, the appearance of Queen Elizabeth on our dollar bills raises a political issue these days around. Remember it all started with a simple question, but the author has to change it completely to illustrate whatever the purpose is.

French courseware.

Early efforts for writing new courseware include an introduction to data processing with BASIC programming. It has been taken by a lot of students, potentially more than a thousand per annum.

Another is a sampling techniques lesson, with emphasis on size and representativity of the samples. It is the result of a one man-year project of the author of these lines with professor Claude Angers. Students are at the master's degree in public administration. Publication has been asked for this lesson, a "première" on the Québec PLATO system.

New material is also developped in various areas of chemistry, physics and mathematics.

Telecommunications.

No doubt about it: telecommunications are a problem over a province nearly as large as Mexico but less populous than New York city. Quality of lines is rather good. The only problem is their cost! The worst case was between the 500 miles distant cities of Rouyn and Québec: the only rate available was the DDD at \$30./hour! Here at "Ecole Nationale d'Administration Publique" we used to pay \$300./month for a dedicated line between Montréal and Québec (150 miles) and this was with government preferential rate. A long awaited

- 38 multiplexing fortunately slashed the cost in half. We now hope for another reduction with packet switching. The Québec PLATO system communicates with the international PLATO community through Minneapolis. Lessons and notes are transferred over a \$18,000./year link. Minneapolis in turn communicates with Brussels every day so there is a one day time-lag for Québec-Brussels transfers. Reduced satellite communications rates should give us in the near future a more comfortable direct link to that french system.

Bibliography

- (1) Université du Québec. Vice-présidence aux Communications. L'implantation de PLATON à l'Université du Québec. Août 1977.
- (2) Université du Québec. Vice-présidence aux Communications. PLATON, un système multimédia d'apprentissage. Août 1977.
- (3) Dumas, Pierre. Guide d'organisation du projet PLATON à l'Université du Québec. Avril 1978.
- (4) Dupuy, Robert. PLATON - Guide de l'utilisateur. Novembre 1978.
- (5) Schwab, Wallace; Dupuy, Robert; Dumas, Pierre et Reid, J.B. Le Guide du médiateur: introduction au langage PLATON. Janvier 1979.
- (6) Schwab, Wallace. Le Guide du traducteur. Document de travail. 1ère édition prévue pour mai 1979.
- (7) Université du Québec. Vice-présidence aux Communications. Adaptation française de PLATON. Août 1977.
- (8) Université du Québec. Vice-présidence aux Communications. PLATON: Bulletin de nouvelles.

THE WORD-PROBLEM STRAND IN MATH DRILL USING A MINI-SYSTEM

G. Lawrence Hopkins
The Masters School
Dobbs Ferry, New York 10522

A description of a newly expanded Strands package for a mini time-sharing system, emphasizing the unique significance of the WORD-strand beyond Mathematics drill and the 3 vital functions performed by the strand. Described are curriculum, range, vocabulary, flexibility, use, typical examples, grade-level (K-8) suitability, appropriate use, necessary hardware and storage requirements. Also mentioned are observations from successful field-testing, the equilibrium effect, suggested approaches, adaptability, overview and future potential.

Introduction

The Strands-concept for Math drill is not new. Many educators are aware of the usefulness, power and flexibility not only of computerized drill but also of the Strands concept itself.

This paper will review the general Strands approach and then describe a flexible, full power package which was developed by teachers from their daily teaching experience and which operates on a non-dedicated mini-system.

Particular emphasis will be paid to the WORD-problem strand.

The Math-Strands Concept and Package

The "strands" consist of graded drills in a number of areas of Math, and over a spectrum of stages from Kindergarten to roughly Grade 8. A student is enrolled in the program by the teacher at any pre-determined starting level, with a teacher-optional stopping level, and from then on the computer gives the student appropriate questions, selecting randomly from each of the "strands" which are appropriate. Each question type is generated with different values of the variables each time it is used. The computer keeps separate records of the student's progress on each strand separately, so that in effect a student might during some particular session work on many different strands, each possibly at a different progress level, and the teacher can at any time find out not only the student's progress in each strand but also when the work was done.

If a student does well on a particular strand in a session, then the next session will drill the student in that strand at a level 1/10 stage higher. If the work on the strand is merely satisfactory, then the next session is at the same stage-level, and if the work is poor,

then the level is shifted downward 1/10 step.

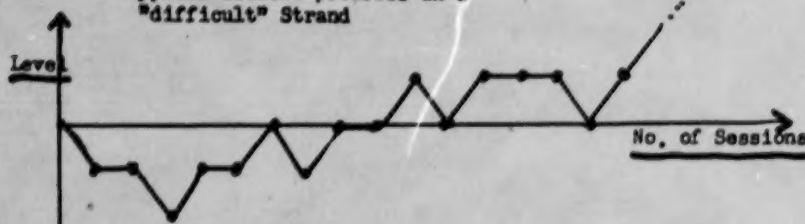
It will be noted from the table that although a rich curriculum is provided, no one strand encompasses the entire spectrum. It should also be noted that the stages 0.1 - 7.9 are arbitrary and do NOT necessarily correspond to the customary year grades 1 - 7.

MATH STRANDS

Strand	Begin	End
ADDITION	0.1	5.9
SUBTRACTION	0.1	5.9
MULTIPLICATION	0.6	7.9
DIVISION	3.5	7.9
NUMBER CONCEPTS	1.0	7.5
INEQUALITIES	1.5	7.4
FRACTIONS	4.0	7.9
DECIMALS	4.0	7.9
NEGATIVE NUMBERS	6.0	7.8
EXPONENTS AND ROOTS	5.0	7.5
LAWS of ARITHMETIC	3.0	7.9
PERCENT	6.3	7.9
WORD PROBLEMS	1.3	7.9

It is common practice to start students in at a point somewhat below their anticipated competence levels for both review and a running head start. The first few sessions generally see very rapid morale-building progress up through several 1/10-stage steps until a slow-down occurs. The student is then likely to spend several sessions in a strand at the same stage, or possibly drop down for a while fractionally, before progressing upward again. Most commonly a student will make several ups and downs across each stage-level before going upwards, but the general trend is upward, even in a "difficult" strand.

Typical student progress in a
"difficult" Strand



Complete management and reporting facilities are available for teachers and lack of student progress in a strand is a signal for special attention.

Each strand "terminates" when a student attains its "end" level and the drill package terminates at stage 7,9, but the teacher may arbitrarily set the limit beyond which the student will not be taken for the strands as a whole.

Teacher options include:

starting level
ending level
session length
"acceptable" score for a session
final end-of-session score printout for student
permitting student to continue work at the end level of a strand

Educational Value

It is important to point out that this type of drill-and-practice is not teaching (except in the limited sense that any homework drill teaches) and does not replace any teacher or offer any panacea; however, the strengths offered by computerized drill are well known and that case need not be argued here.

Ideally, what I would like to see for myself as a classroom Math teacher is that when each day I finish work with my class in the classroom I would have ready at hand for them a special and different out-of-class assignment -- a purple sheet handout, if you will -- custom designed and fitted for each student, and that each day for each class! The "Strands" system provides a differentiated, flexible proven and powerful supplement to class instruction. The strands do not do the instructing -- the teacher does -- but they are highly individualized drill & practice and provide enormous out-of-class support for the teacher, doing much the same thing for a student as having a teacher sit down alongside while the homework or drill exercises are done, and no text or exercise book could provide such patient flexibility or almost endless variety.

In our school we have heard students call the computer the "tireless tutor", and we believe that our effectiveness as Math teachers

is substantially multiplied.

The Word-Problem Strand

Purpose

Twelve of the present 13 strands deal with computational, manipulative, mathematical techniques and skills and facts. The 13th or Word problem strand is substantially different in nature and scope.

For numberless reasons students have traditionally had difficulty with verbal or "story" problems, and only a few of the reasons need be mentioned explicitly here. Every experienced Math teacher has grappled with this problem of "teaching" kids to be comfortable with word problems, and all too often with poor results.

In our experience in our school we find a very real difference in the ability of students to handle the following two problems:

- (A) $\square + 5 = 8$; Fill in the box.
- (B) Margaret has 8 candy bars. If Mary buys 5 more candy bars for herself, she will then have the same number as Margaret. How many candy bars does Mary have now?

These two problem types are familiar and are almost identical mathematically; they are really two different versions of the same problem. However, these distinctions should be noted:

- i) In version (A) the arithmetic process is indicated by the "+" sign, whereas in version (B) no clue is given except through such code-words as "more".
- ii) In version (B) the student must sort out the named people and make sure who is who and who has what, and then associate them with quantities, and then must order those quantities, etc.
- iii) In version (B) the student must read what is said, and very carefully.

There is obviously a tremendous difference between the versions and we Math teachers spend much conscious time and effort on guiding our students to look for and interpret code-words such as "more than" or "less than", and to write

mathematical sentences which parallel the English sentences of the word problems, and so forth. Probably none of us ever feels we do as much as we should to help students develop these skills, and that we never find enough time or suitable graded verbal drills, etc.

I, myself, am one of the many who feel that comfortable ability in handling the verbal forms is essential preparation for the pay-off in later years when such "heavy" areas as Science and many other applications are encountered. I am professionally confident that any teacher who has taught Calculus will agree.

I often encounter critics who maintain that learning the manipulative techniques and procedures of arithmetic and algebra is the "main game" and that the word situations found in textbooks are artificial, unrealistic, contrived, irrelevant and almost useless. This criticism has some partial validity for many texts and it is not possible to go into a major discussion of all the concerns that are raised, but it would also be possible to make an impressive case for the proposition that working well with words is a very strong, and maybe even necessary, step toward conceptualizing.

We educators and computer people also know from ample experience that being able to articulate a problem or state it clearly is a critical step toward the problem's solution.

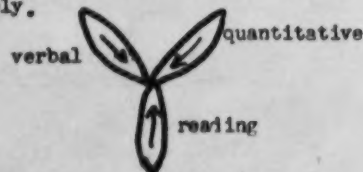
To the objection that only a special few students will ever need these mathematical skills it may be fairly rejoined that the possession or lack of possession of these skills is precisely one of the criteria which ruthlessly sort out and select those "special few"!

This seeming digression comes from a profound conviction that it is the cultivation of these verbal skills over and above the quantitative and manipulative ones which can make the difference for a student between failure and meaningful mathematical success.

The Unique Position of the WORD-Strand

Whatever the causes, we are uncomfortably aware that there is now responsible confirmation of a nationwide decline in computational skills, reading skills, and so forth.

So it can be seen that the verbal problems of Mathematics and thus the WORD-problem strand lie at the interface between verbal and quantitative skills and also have the third function of practice in reading. The combination then transcends mere Math drill; working with word problems hits these three things squarely and uniquely.



Content and Curriculum

The WORD-problem strand begins at a stage roughly comparable to the middle of Grade 1. Earliest exercises involve counting and reading of very simple instructions and contain little word material. The vocabulary, reading difficulty level and mathematical ideas are introduced gradually in stages (marked in tenths) consistent with standard curricula. For example, in the material which is suitable for early Third Grade the question types which require addition or subtraction are carefully built to avoid regrouping (the "borrowing" and "carrying" of yesteryear

Peter collected 38 newspapers.
15 of them had torn pages. How many of them did not have torn pages?

(Note: Integers without re-grouping)
Circled numbers are generated answers.)

At each step along the spectrum the student encounters a variety of "questions" suitable to that step. If a student stays at one stage for a while, the same question-type may appear more than once, but each time with different number quantities (as circled). In addition to this, a question-type is likely to appear in several process variations as follows:

Joe has only $\frac{1}{3}$ as many patch emblems on his jacket as Mike does. If Mike has 18 patch emblems, how many does Joe have?

OR Joe has only $\frac{1}{4}$ as many patch emblems on his jacket as Mike does. If Joe has 12 patch emblems, how many does Mike have?

(Note: Practice with fraction concepts, and also the process change, making it difficult to "memorize" a problem involving Joe and Mike.)

Some question-types are used at more than one level, but many are unique to a single level both to avoid staleness and because of the need for different vocabularies and concepts.

One typical stage might for example have eight question-types assigned to it. Six of these might each involve only one process such as addition or division, while the others might require two or more processes each.

Mrs. Jones bought 2 quarts of milk for \$0.60 each and 3 cans of fish at \$1.10 each. How much change should she receive from a \$10 bill?

(Note: Several processes in a somewhat involved but real situation.)

42 Equilibrium Effect

Very frequently a student may encounter only one or two WORD questions in a typical session, answer them both correctly, and thus go on to the next higher level for the next session. The next session may go poorly, however, with a consequent dropdown. We see again here a so-called "equilibrium effect" where if a student rises to a too challenging stage, he/she then falls back down, or conversely rises from a too easy stage, thus reaching temporary equilibrium at the suitable stage-level. The entire process is thus both flexible and dynamic.

The Present Version

Our present version was developed by the Riverdale Country School as an extension of older more rigid systems, with our own school, Masters, preparing the material in the Percent and Word problem strands. The new version incorporates teacher control features and student material which are the direct result of almost five years' daily testing with our classes.

Not only is this new Strands package efficient enough to run on a mini-processor, but the processor and system are not monopolized by the Strands and can handle other normal time-sharing tasks concurrently. In the other direction, the package is equally usable on a large system to handle up to 128 terminals and thousands of students.

Hardware

The system presently in use is a DEC PDP-11. The disk storage requirements are about 205000 bytes for the package, plus 160 bytes for each student enrolled. The run-time system is RSTS/E.

Future Plans

The present curriculum covers up to the beginning of First Year Algebra. We are now planning the extension of the WORD strand up into the verbal materials of Elementary Algebra, and we contemplate also the development of a completely new strand dealing with the skills and techniques of Algebra. This Algebra Strand is for the present only exploratory but would begin at what is stage 8.0 on our scale.

An additional version is also in the development stage and will run on a still smaller processor.

A Suggested Procedure

One possibility is to assign students in a class to do a certain number of timed sessions outside of class each week, possibly cutting down on other regular "homework". If hardcopy

terminals are available, it is advisable to start all strands' work on them so that all printouts may be handed in to the teacher for careful attention during the early stages. Later on CRTs might be used if available since records of satisfactory sessions and progress in each strand are instantly obtainable.

Experience at the Masters School

We have found that the "Strands" approach is a priceless supplement to our math classwork, and is well received by students. We have only three terminals and our teachers require all pre-algebra students to do strands work several times a week as part of "homework", but we find increasingly that the Strands provide a much needed support and refreshing in basic skills for our Algebra and Geometry students. We have, therefore, come to use the Strands for most of these classes, too.

We customarily start students at a stage level considerably below where we feel they ought to be, for both a running head start and to provide a non-threatening introduction at a level where some instant success should be assured. We have found, not surprisingly, that students generally rise rapidly in some strands but sink initially in others. Since the inauguration of the WORD problem strand the students in the 7th, 8th and 9th grade classes have sunk conspicuously more in this strand than in others. In fact, many students did not stabilize until reaching down to about 5th grade work. Since our school is 7 - 12 grades, this told us something diagnostically about students in general and about the specific tasks we had cut out for us. This signal became even more marked for us when we saw that these same students generally understand associative principles and the like and can manipulate and compute reasonably well.

Another surprise has been that so many students at all ages cannot figure the sale price on an item marked down, say, five or ten percent, or cannot figure their percent grade on a test where perhaps 27 answers are correct out of 40.

We have seen that additional class and individual work with these students, as well as continued Strands practice during the school year, has brought these students up markedly. With our own particular choice of about two Strands sessions for a student per week (due to our small number of terminals) students get less than what we would consider ideal exposure to the Strands, or between 50 and 60 sessions. What with the initial "bottoming out" at an equilibrium point in the early stages and with the zig-zag rise in any strand throughout the year, we would like more terminals to give us more sessions per student per year. However, even so, we expect and commonly get increases of about 2 full stage-levels in most students before the year runs out.

It should be mentioned here that we also use a considerable variety of other special purpose computer drills for classes and individuals,

- (1) This dynamic adjustment and "equilibrium" combination operates in each strand.

for example in integers, but the overall flexibility and dynamic capabilities of the Strands system adjusting to and nudging each student along give us a tremendous power and strength not otherwise attainable.

One other thing in our experience is the increasing number of students who, as Juniors or Seniors, are not enrolled in Math classes and who are discovering that the Strands drill is a valuable refresher to "get their hand back in" before Scholastic Aptitude Tests or the like. We do not urge this since we feel that there are other ways of readying oneself for such things, and we have but 3 terminals, but we are struck by the number of such students who ask to use the Strands and who tell us they are helped.

Conclusion and Observations

In the experience of the schools on our small system the Strands, then, are a valuable tool for grades K-12, even though designed for K-8. This immediately suggests a possible application for the Junior College in search of a tool for helping students in need of remedial work in basic math skills.

Certainly this teacher hopes never again to teach Math in a situation where the Strands package is not available, and looks forward eagerly to the day when the package will be enhanced and extended.

Further Examples

If one addend is 12 and the sum is 36, what is the other addend?

Note: Attention to vocabulary; the process is not "clued" by the numbers (a student who looks for numbers which obviously "divide" may try division.)

It is 8:45. How many minutes does Mike have to wait until it is 9:22?

Note: a useful skill emphasized

Red cloth costs \$5.40 for 2 square yards but blue cloth is \$8.50 for 3 square yards. Which is the more economical buy, red or blue or neither?

Note: non-trivial consumerism

Mary earns \$150 per week at her present job but spends \$60 of it on food and entertainment. In lowest terms (numerator/denominator) what fraction of her income does Mary spend on food and entertainment?

Note: concepts of fraction and reduction to lowest terms

The distance around a rectangular yard is 180 meters. If the length of one side is 40 meters, what is the area of the yard in square meters?

Note: A large variety of topics and skills may be dealt with at various levels of sophistication, both verbal and mathematical.

INSTRUCTION / LEARNING APPLICATIONS

OTHER THAN CAI

INCLUDES RESEARCH/EVALUATION

AMUS: A SCORE LANGUAGE FOR
COMPUTER-ASSISTED
APPLICATIONS IN MUSIC

by

W. Kenton Bales

and

Joan C. Groom

A versatile, cost-effective microcomputer-based sound generation and delivery system has been developed at North Texas State University (NTSU). The Automatic Music System (AMJS), designed and constructed at NTSU by Dan W. Scott, provides simple flexible encoding for musical notation, and real-time generation of up to eight voices. The NTSU School of Music has utilized a prototype to produce a computer-assisted instruction package for aural skill training, and computer science courses use AMUS to explore musical applications of microprocessor technology. This paper describes the AMUS hardware and software and their applications at NTSU.

Contemporary developments in the use of computers in the arts and humanities have necessitated a re-evaluation of our concept of specialization in education. No longer may the educator or researcher concentrate on a single area only: now he must show how this area relates to other divisions within his own discipline, as well as to other subjects. This structuralistic approach (Piaget, 1977) justifies a return to the Renaissance concept of the multidisciplinary, humanistic personality, exemplified educationally in the awarding of multiple advanced degrees. (Science fiction writers such as Philip Jose Farmer, in his "Riders of the Purple Wage", and Jonathan Brand, in "Encounter With a Hick" speculate on the type of society that might require multiple Ph.D.'s.)

Methods of easing the strain on already overloaded curricula can be realized with enlightened educational planning. The use of computer-assisted instruction (CAI) will familiarize students with computers early in their careers. Instruction in mathematical analysis of music will sharpen the students' use of mathematics. Exposure to electronic musical instruments (as well as to traditional instruments) will increase the students' awareness of electronics. By teaching the basics of disciplines such as mathematics and electronics as a part of musical skills, musicians will become conversant with them, and therefore more capable of efficiently setting up computer based music systems.

Specialized equipment that is efficient, flexible, and cost-effective is necessary for these educational purposes. The Automatic Music System was developed at North Texas State University by Dan W. Scott with these considerations in mind.

THE AMUS HARDWARE

The Automatic Music System plays multiple-part, note-oriented music. Parameters such as

amplitude, pitch, tempo, and articulation are specified by the user, who provides a text (the "score") written in ordinary typewriter characters by means of a computer terminal (Lear-Siegler AIM 3-A terminals are used at NTSU). This score is translated by a microprocessor into data for the digital music instrument, Musor. The score may be typed from the terminal, stored on cassette tape, stored or generated as text by a remote computer (the "host computer"), or it may be the text in a CAI system which uses the host computer.

The host computer at NTSU is a 32-port Hewlett-Packard model HP-2000 Access BASIC timesharing system. A Southwest Technical Products Motorola M6800 microcomputer is the processor for the Musor instrument. The instrument is accessed as if it were a location in memory of the M6800, making a hardware interface all that is necessary for implementation. This design feature makes possible the use of microprocessors other than the M6800.

The Musor instrument consists of a card cage, edge connectors, wiring assembly, a control panel, a 5-volt power supply, and a plus-minus 15-volt power supply. The instrument is connected to the microprocessor by a short cable. Four types of 6 1/2 inch printed circuit boards make up the Musor instrument: I (the computer interface), D (the digital signal clock generator), A (the digital audio generator), and M (the audio mixer and output amplifier). The minimum configuration necessary for monophonic music includes one board each of I, D, A, and M. The maximum experimental system consists of one I, two D, eight A, and one M, a total of twelve boards.

The Musor subsystem contains up to eight individual waveform generators (boards A). Their outputs are mixed into a single analog output. Each generator is the same design, but waveforms may be defined differently for each voice. The final output can be sent to headphones, to an amplifier, or to a tape recorder. (See Bales and Hamilton, 1978, for a more detailed description of the system hardware.)

In order to handle a variety of input symbols and tasks, the AMUS program has three modes of operation: text mode, score mode, and terminal mode. Certain pre-defined commands change the mode, and therefore change the interpretation of input.

In text mode, anything typed on the control terminal is sent to the host computer, and anything received from the remote computer is displayed on the control terminal. AMUS takes no action until it reads a question mark (?) or a percent sign (%). A question mark causes the score which was last read by the computer to be played. The user may cause a replay at any time by typing a question mark or by causing the host computer to transmit a question mark.

The percent sign signals AMUS that the next characters should be some kind of directive. If no valid text mode command is found, the program takes no action and continues to read and display characters in text mode. If the percent sign precedes "S" or "SC", AMUS deletes them from display and enters score mode. In score mode the input characters are translated into Musor control data for later performance. Score characters received from the host computer are not displayed to the user, unless he so directs.

To exit score mode, a question mark can be typed, which begins performance of the score and returns control to text mode. The command "%E" also transfers control to text mode, but does not play the score.

Terminal mode is entered from text mode by means of the "%X" command. In terminal mode, the system reacts as a computer terminal. All texts, including text and score mode directives, are sent from the control terminal to the host computer without processing. Terminal mode is exited by typing ESC (escape), BRK (break) to return to text mode. Control may be directed to the MIKBUG program of the M6800 by the "%." command, in order to work in M6800 assembly language.

The AMUS score language, used in score mode, consists of several note specifiers. These are described in Table 1.

NOTE SPECIFIERS

0 1 2 3 4 5 6 7	Voice number designator. Must be first character of note. If missing, "1" is default.
A B C D E F G R	Pitch name designator. Represents musical pitch by note name. "R" indicates a rest.
# + P	One semitone sharp.
X	Two semitones sharp.
B -	One semitone flat.
BB --	Two semitones flat.
Z	Natural. Accidentals normally follow pitch. When more than one is used, the effect is cumulative. They do not apply throughout the measure, they affect only the note with which they are associated.

U D

Octave shift designators.
Up or down, respectively. Shifts current note and all following notes up or down one octave. If more than one is used, the effect is cumulative.

W H Q E S T 4

Duration specifiers.
Whole, Half, Quarter, Eighth, Sixteenth, Thirty-second, and Sixty-fourth notes, respectively. Following pitch, accidental, and octave shift (if latter two are present). They may be combined with additive effect.

. 3

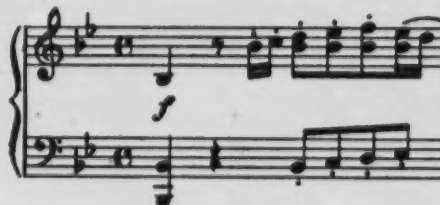
Dotted value and triplet specifier.
A dot following a duration makes the note 3/2 its original value. A three after a duration makes the note 2/3 its original value.

L N ,

Articulation specifiers.
Affects the four note-fragments and their amplitudes.

TABLE 1.

Figure 1 is an example of music encoded in the AMUS language.



```
%S %K=2- %MM=100
3RD DD 2BD 1BDQ 3R 2R 1RE 1BUS 1CUS
3BU 2BUU 1DE 3CU 2B 1EE 3D 2B 1FE
3E 2B 1ES 2DU 1RS /
```

FIGURE 1.

The AMUS program, stored in the M6800 as four thousand bytes of Read Only Memory, is primarily devoted to score translation. The input (from terminal, cassette tape, or another computer) is translated from ASCII standard characters into an internal score code containing all of the information needed by Musor. This information is stored in variable-length blocks in a predefined format.

The duration timer is an interrupt-driven program which counts down in units of 568 microseconds until the 16-bit count, set by the playing program, reaches zero. When zero is reached, the next block of internal score information is sent to Musor, and the timer is set to the new duration value. Performance can be concurrent with the input and translation processing, allowing the beginning of the score to be played while the remainder of the score is being read and translated.

A tone produced by the Musor instrument consists of as many as four fragments. Each fragment

is a constant frequency with a constant amplitude. The four fragments are analogous to the attack (two fragments), steady state, and final decay of the envelope of a note. Four constant amplitude levels are available for each fragment, plus a fifth non-constant amplitude which decays over a defined period of time.

APPLICATIONS OF AMUS AT NTSU

AMUS is currently in use as the core of an extensive CAI facility in music theory at NTSU. The curriculum was developed by Rosemary N. Killam and programmed by Richard L. Hamilton. Some seventy-five drill and practice lessons are currently in use covering the first year of auditory training, and curriculum for the second year is under development. Courseware coordinated with Robert W. Ottman's Elementary Harmony and Advanced Harmony texts constitute the majority of lessons. Lessons in chord identification, intervallic, harmonic, rhythmic and melodic dictation are available.

For the music student who is unfamiliar with CAI, the AMUS system offers immediate exposure to two computer systems: the HP 2000 and the M6800. Early lessons are designed to minimize the inevitable technological shock that some students experience. The apprehension of the "Big Brother" that dictates practice in aural training fades as the student familiarizes himself with the equipment. The student does not need to know any programming language in order to use AMUS, but may take classes at NTSU which present detailed information on the hardware and software design of the system. These classes have been well received by students and administration alike, and provide the project with trained personnel for continuing research and development.

Because of the large number of users at NTSU (some 800 students and 25 instructors), the programs allow for simple score encoding and lesson development. The curriculum driver in use at NTSU was designed and developed by Richard L. Hamilton. Its format is similar to the of PILOT, but it includes additional features necessary to music drill and practice (Hamilton and Scott, 1978). This simple, flexible lesson coding encourages skeptical educators to use the computer as a tool for expanding their instructional concerns, helping to create a body of computer-literate instructors that don't shy away from CAI.

The system's link to a host computer makes it useful for applications in computer-aided composition. Any number of compositional routines executed by the host computer can be used to generate score code for the Musor instrument. "MONODRAM: for Synthesized Sound and Player", by W. Kenton Bales, uses the AMUS both as a score generator and as a digital sound source (Bales and Hamilton, 1978). MUSIC, also by Mr. Bales, is a continually developing program for computer generated composition. The program is sectional, and each block defines individual note values for each section.

The AMUS system exposes the composer to interactive digital synthesis. He may enter all or any portion of his score, and hear it immediately by typing a question mark. This exposure to digital synthesis techniques increases the composer's awareness of the computer as a tool for the devel-

opment of his craft. A knowledge of a programming language (in this case, BASIC) is necessary for successful implementation of score generating techniques. This introduction to computer programming will prepare the composer for more advanced digital music applications on other systems as well as for more advanced compositional techniques.

The AMUS system is beginning to lend itself to experiments in psychoacoustics. The 800 students who use the system provide control and experimental groups, and the system's powerful data collection programs can store and retrieve all the information necessary for rigorous study. Currently under way is a psychoacoustical experiment designed by Leslie Gay, which studies the effect of pitch on the perception of time in music.

Daniel Peak is studying the feasibility of using the system to teach composition to grade school children. Preliminary results show that children respond quickly to AMUS. Only a small amount of time is necessary in order to teach the children to use the computer, and the individualization provided by AMUS surpasses that of most classroom situations. Experiments that are designed for senior citizens (an embarrassingly neglected group) are also in the planning stages.

The multiple applications of the Automatic Music System at NTSU has led to the formation of a Computer Music Research Consortium (CMRC). Progress reports and seminars in computer applications constitute the bulk of weekly meetings, and resources are pooled in order to expedite effective use of research time. The ten members include faculty, graduate students, and undergraduates, all capable of defining and implementing computer-based educational systems based on their experience with AMUS at NTSU.

CONCLUSIONS

At present, teams of specialists are necessary in order to implement most computer-based music facilities. Electrical engineers design circuits and equipment, programmers implement routines to present musical ideas, and mathematicians formulate precise methods for computing musical parameters as they are defined by musicians. The battery of necessary personnel who do not speak each other's language (musicians seldom speak electronics and vice versa) increases the possibility of error in implementation of any project. The obvious solution is to educate students and instructors in the area of computer-based instructional systems. The curriculum would include liberal doses of mathematics, electronics, computer science, and psychoacoustics, as well as traditional musical training.

In the future, when one applies for a job, the interviewer may well assume that the applicant can read, write, and compute. The most efficient use of computer-based educational systems will come from users who possess proficiency not only in traditional skills, but also in areas that presently appear to be unrelated or superfluous. If these users are to set up effective CAI installations at their institutions, their knowledge of computers will tell them what kind of program design is possible. Their experience with electronics will help them choose equipment best

50 suited for their purposes, while their programming techniques will allow them to modify and improve the system as the need arises. Mathematical skill may be taxed by attempts to balance inadequate budgets, but administrators will appreciate an understanding of cost-effective implementation. New systems and new software packages will simplify the monumental task of implementing CAI installations. The development of systems like AMUS may help foster the rise of a new kind of educator, capable of meeting the demands of computer-based curriculum design and development. These educators will be well versed in several related disciplines, pragmatic, creative, and sensitive to the needs of their fellow humans: in short, members of the New Renaissance.

REFERENCES

- Bales, W. Kenton and Richard L. Hamilton and Dan W. Scott. "Computer-Aided Composition and Performance with AMUS". Proceedings of the 1978 International Computer Music Conference, Evanston, Illinois.
- Brand, Jonathon. "Encounter With a Hick". Dangerous Visions, Harlan Ellison, ed., Signet Press, New York, 1967.
- Farmer, Philip Jose. "Riders of the Purple Wage". Dangerous Visions, Harlan Ellison, ed., Signet Press, New York, 1967.
- Hamilton, Richard L. and Dan W. Scott. "A New Approach to Computer-Assisted Instruction in Music Theory". Presented at the 1977 Conference on Computers in the Undergraduate Curriculum, Denver, Colorado.
- Killam, Rosemary N., W. Kenton Bales, Richard L. Hamilton and Dan W. Scott. "AMUS: The Computer in Music Instruction". Presented at the 1979 Texas Music Educators' Association Conference, Fort Worth, Texas.
- Piaget, Jean. The Development of Thought: Equilibrium of Cognitive Structures. A. Rosin, trans., Viking Press, New York, 1977.

AUTOMATIC PROGRAMMING: AN
EDUCATOR'S AID FOR
COMPUTER-ASSISTED TEST
CONSTRUCTION (CATC)

Ronald W. Collins and Steven J. Duff
Eastern Michigan University
Department of Chemistry
Ypsilanti, Michigan 48197

Computer-Assisted Test Construction (CATC) has been widely used in the sciences, both for generating exams and problem sets. The most popular approach has been to use a large database of categorized questions/problems, from which the computer makes appropriate, but randomized selections. Our FORTRAN program PROBGEN offers, however, a more sophisticated approach to CATC, even for faculty members who cannot program. By answering a series of questions on-line, the faculty user defines his/her problem to PROBGEN, which in turn writes another program (i.e., automatic programming) to both create and solve a series of specific, non-identical problems based on the general model. Each student may then be given a test/assignment with unique but analogous problems.

The most standard component of all teaching-learning situations (i.e., courses) seems to be the administration of assignments and exams to the students, either for the purpose of stimulating study or of evaluating student performance. Since learning, or at least mastery, of any subject often involves repetitive cycles of study and testing, there is also widespread agreement among many instructors that individualized, repeatable tests are desirable. We have therefore concentrated our software efforts in the area of computer-assisted test construction (CATC). To meet the criterion of easy transportability, we have written our "PROBGEN" (problem generator) programs in standard FORTRAN.

According to Lippey¹, the computer can support test preparation in five ways: (1) item banking, (2) item generation, (3) item attribute banking, (4) item selection, and (5) test printing. In addition, the computer can assist in grading by generating answer keys and/or by actually scoring the exams; however, scoring is basically an administrative function of the computer and as such is not the principal focus of CATC.

There are two distinctly different approaches to CATC. The first is based on the ability of the computer to facilitate information retrieval (hereafter referred to as CATC-R), and the second method involves the computer in a generation role (called CATC-G). In other words, in CATC-R the computer merely processes a categorized database of complete questions, with or without answers, but does not actively create or add to the content of the individual questions themselves. This technique requires an extensive inventory of questions in machine-readable form ("item banking").

In chemistry, the two best known CATC-R systems are those of Moore² (based on an earlier system developed by Prosser³) and Willis⁴. In each instance there is a main program which controls the processing of the database of questions. The questions are classified according to topic, degree of difficulty, etc. It is important to stress, however, that "what goes in, comes out"; i.e., the computer does not solve the problems as it randomly selects from the available categories. Answers will be printed for the tests only if the problems had been solved previously and the answers had been included in the database.

In CATC-G the computational power of the computer is employed to assign values to the variables within problems, and to solve the resultant, unique problems ("item generation"). In this instance there is no need for an inventory of questions. Instead, one program or subroutine must be written for each type of problem. The author determines the format or "skeleton" of the desired problem, and then writes a program which will then generate values randomly within appropriate limits and "fill in the blanks" in the problem. Concurrently, the computer solves the problem and provides the correct answer. In chemistry, the most successful CATC-G system is that developed by Johnson⁵ at the University of Pittsburgh, which currently consists of approximately 400 subroutines (each being a question prototype) which are in turn executed and controlled by a main program.

The implementation and expanding use of a CATC system can obviously be quite time consuming. In the case of CATC-R, a lot of time

will be spent either in writing new questions or in reviewing and/or classifying questions obtained elsewhere. On the other hand, in CATC-G much time will be spent in writing new programs, one per problem type.

Assuming programming expertise, the CATC-G approach appears to be a more creative computer-based method. On the other hand, it is obvious that many instructors use CATC-R simply because they do not have the required programming expertise. We therefore concluded that what was really needed was an "automatic" CATC-G system for non-programmers. In other words, the computer itself should serve as the program/subroutine writer.

Our PROBGEN program has thus been designed so as to utilize "automatic program generation", or APG. The faculty user never (consciously) writes a single line of program code, but he/she is required to abide by the rules of FORTRAN for variable names and arithmetic operations. In interactive mode, PROBGEN prompts the instructor to provide a generalized solution to each problem along with any major restrictions which might apply, the limits on all variables within each problem, and the exact wording for the text of each problem. After this question-and-answer dialog, PROBGEN then writes the actual program (a CATC-G module) which will create the desired number of non-identical problems of the defined type. Eventually, each test or assignment can consist of one or more problems of as many types as are input to PROBGEN.

In our earlier paper⁶ on this APG approach to CATC we described the first edition of the PROBGEN series (now referred to as PROBGEN I) of programs. Our current paper describes PROBGEN II, a much improved version of the earlier software. The efficient use of PROBGEN II still demands, however, that the instructor carefully organize each problem into a generalized format beforehand; i.e., before accessing PROBGEN II on-line.

In outline form, the steps involved in using PROBGEN II are as follows: (steps 1-9 off-line)

1. Define the type of problem, including the variables (A, B, C, ...) whose values will be given, and the variable (X) whose value will be calculated by the student.
2. Combine and/or arrange the necessary equation(s) to produce an "answer equation(s)" of the form $X=f(A, B, C, \dots)$.
3. For all non-subscripted variables, A, B, C, ..., specify the upper and lower limits, between which single values can be randomly assigned by the program.
4. For all subscripted variables, D(I), E(J), F(K), etc., specify the limits plus the number of values to be randomly assigned.
5. For each variable select a FORTRAN code (variable) name which is consistent with its integer or floating point character.

6. Specify any forbidden values for each variable or forbidden relationships among different variables.
7. Compose the exact wording of the problem as you wish it to appear on the test or problem set.
8. Repeat steps (1) - (7) for each additional problem.
9. Compose the heading for each test and also determine how many problems of each type are to be included per test, and how many tests are desired.
10. (On-line) Access PROBGEN II via terminal and input the appropriate equation(s), values, and text for each problem type upon request; i.e., respond with the proper input to each question from PROBGEN II. Check all input when it is printed back, in order to catch typographical errors before they possibly suppress the execution of the program.
11. Go to the computer center to pick up the tests which are printed on the high-speed line printer.

Although the procedural steps involved in using PROBGEN II are quite similar to those previously used for PROBGEN I, PROBGEN II has greatly expanded technical capabilities (Table I). In order to better illustrate these features, two sample chemistry problems have been chosen for a demonstration run of PROBGEN II. The problems are: (1) a strong acid-strong base titration problem, and (2) a combined gas law problem. The concept and format for both problems are given in TABLE II. A sample test consisting of one of each of these two demonstration problems is shown in Figure 1. The answer sheet accompanying the problems as generated by PROBGEN II is shown in Figure 2.

Demonstration problem number 2 involving the combined gas law equation illustrates the use of the PROBGEN II multiple-choice option. Following the input of the correct answer equation, the instructor provided three "logical wrong answer" equations. These three equations were based on three common student errors: (1) the failure to convert from degrees Kelvin to degrees Celsius; (2) inversion of the pressure factor; and, (3) inversion of the volume factor.

For the sake of brevity in this paper, just two sample chemistry problems have been used to illustrate the capabilities of PROBGEN II. It should be obvious, however, that PROBGEN II could be applied in other disciplines equally well. Any numerical problem which can be systematically generalized into an answer equation(s) may be processed via PROBGEN II, irrespective of whether the topic is physics, biology, or economics.

In terms of limitations, PROBGEN II is applicable only to the generation of quantitative problems. It cannot be used to create theory or discussion questions, and thus PROBGEN II cannot produce as wide a range of test types as can a CATC-R system.

On the positive side, PROBGEN II can be employed to create new problems by an instructor who is a non-programmer whereas additions to the Johnson CATC-G system require considerable programming expertise. Thus, PROBGEN II not only generates individualized, non-identical problem sets or tests for student use, but it permits the faculty member to easily and quickly (usually 15-20 minutes usage time) individualize the style of the problem according to his (her) pedagogical preferences.

In conclusion, PROBGEN II represents a significant breakthrough in making CATC technology available to all educators. Although it does not allow the user to communicate completely free of programming syntax, PROBGEN II is an efficient APG system which does indeed rely on the computer itself to write the programs involved. For more information on the state of the art in the field of APG, consult the excellent review article by Prywes.⁷ The present FORTRAN version of PROBGEN II is available upon request (on magnetic tape) for a very modest fee covering only the handling and mailing costs.

References

1. Computer-Assisted Test Construction, G. Lippey, Editor, Educational Technology Publications, Englewood Cliffs, N.J. 07632 (1974).
2. "Computer-Generated Repeatable Tests in Chemistry", J.W. Moore, Proceedings of the Conference on Computers in Chemical Education and Research, Northern Illinois University (1971).
3. "Repeatable Tests", F. Prosser, Educational Technology, 13 (3), 34 (1973).
4. "The SOCRATES CATC System at the California State Universities and Colleges", W.V. Willis and O. Seely, paper presented at the First Chemical Congress of the North American Continent, Mexico City, Mexico (1975).
5. "Pittsburgh's Computer-Generated Repeatable Exam System", K. J. Johnson, Proceedings of the 1973 Conference on Computers in the Undergraduate Curricula, p. 199 (1973).
6. "PROBGEN: An Interactive Computer Program for Faculty Use in Creating and Generating Individualized Non-Identical Problem Sets and Tests," Proceedings of the Ninth Conference on Computers in the Undergraduate Curricula, p. 100 (1978).
7. "Automatic Generation of Computer Programs," N.S. Prywes, Advances in Computers, Volume 16, M. Rubinooff and M.C. Yovits, Editors; pp. 57-125, Academic Press, NY (1977).

VARIABLES - type, user control, etc.	(1) "Dummy variables" not needed in answer equations; the program calls for the limits on such variables in a separate step (2) All variables are assumed to be randomized; single value quantities are to be included as numerical constants in the answer equation. Program is less text intensive. (3) Subscripted variables of three types may be used: X(I)-program assigns n regularly-spaced incremental values within specified limits X(J)-program assigns n random values within specified limits X(K)-program accepts n specific values for the variable (4) User can dictate significant figures for each variable via the form of the input for the limits
PROBLEM FORMATS - several parts, multiple choice	(1) Problems may have several parts, with the parts being either inter-related or completely independent of each other. If related, the answer equations may be set up sequentially; i.e., using one answer to calculate the next, etc. (2) The user has the option to use multiple choice format for any problem. In addition to the correct answer, the program provides three wrong answers via: — calculations using three "logical wrong answer equations" input by the user, or — purely random selection of three other values within a range specified by the user
ARITHMETIC CAPABILITIES	(1) Solutions may involve the use of standard library subroutines; e.g., if the answer equation is a cubic, the user may call a Newton-Raphson subroutine for the solution

Table I Capabilities of PROBGEN II

STRONG ACID-STRONG BASE TITRATION PROBLEM		GAS LAW PROBLEM
A titration problem involving five successive additions of the titrant, NaOH, to a solution of HCl	General Concept of the Problem	Combined gas law problem using $\frac{P_1 V_1}{T_1} = \frac{P_2 V_2}{T_2}$
The initial volume, ml _A , of acid The concentration of the acid, M _A The concentration of the base, M _B Five total volumes of base added, ml _{B1} , ml _{B2} , ... etc,	Given (or to be assigned by the program)	All initial values, P ₁ , V ₁ , and T ₁ , The final values for P ₂ and V ₂
Five pH values; one for each total volume of base added	To be calculated by the student	The final temperature, t ₂ in degrees Celsius
(i) if (ml_A × M_A) > (ml_{Bn} × M_B) $\text{pH} = -\log \left[\frac{(\text{ml}_A \times M_A) - (\text{ml}_{Bn} \times M_B)}{\text{ml}_A + \text{ml}_B} \right]$ (ii) if (ml_A × M_A) < (ml_{Bn} × M_B) $\text{pH} = 14.00 + \log \left[\frac{(\text{ml}_{Bn} \times M_B) - (\text{ml}_A \times M_A)}{\text{ml}_A + \text{ml}_B} \right]$	Answer equation(s) and the special condition for each (if any).	$t_2 = \left[T_1 \times \frac{P_2}{P_1} \times \frac{V_2}{V_1} \right] - 273$ P₂ ≠ P₁ V₂ ≠ V₁

TABLE II General Format of the Two Problems Used to Demonstrate PROBGEN II

PROBGEN II DEMONSTRATION

FORM: 7 DATE: 14-JAN-79

PPPP	RRRR	000	8888	GGG	EEEE	A	N	III	III
P P R R	0 0	B B	G G	E	NN	N		I	I
P P R R	0 0	B B	G G	E	NN	N		I	I
PPPo	RRRH	0 0	8888	G GG	EEEE	NNN		I	I
P	R R	0 0	B B	G G	E	NN	N	I	I
P	R R	0 0	B B	G G	E	NN	N	I	I
P	R R	000	8888	GGG	EEEE	A	N	III	III

1. (DEMONSTRATION OF THE USE OF SUBSCRIPTED VARIABLES)
 ASSUME THAT YOU ARE TITRATING A 20.98 - ML SAMPLE OF
 .1965 M HCl WITH .1900 M NaOH. CALCULATE THE PH OF THE
 SOLUTION AFTER THE ADDITION OF EACH OF THE FOLLOWING TOTAL
 VOLUMES (ML) OF NaOH: 7.28, 57.26, 63.22, 63.37, 78.39.

2. (DEMONSTRATION OF MULTIPLE CHOICE FORMAT)
 ASSUME THAT YOU HAVE EXACTLY 92.80 GRAMS OF NITROGEN
 GAS WHICH OCCUPIES 49 LITERS AND EXERTS A PRESSURE
 OF 9 ATMOSPHERES AT 12 DEGREES CELSIUS. WHAT
 WILL THE TEMPERATURE (IN DEGREES CELSIUS) OF THE GAS BE
 IF THE VOLUME CHANGES TO 53598 MILLILITERS AND THE
 PRESSURE TO 4324 TORR? (CIRCLE THE CORRECT ANSWER.)

A. -33 C. 239
 B. 212 D. -125

Figure 1: Sample Test/Problem Set
 Generated by PROBGEN II


```

AAA  N  N  SSS  W  W  EEEEE  HRRR  SSS  H  H  EEEE  EEEE  ITIT
A  A  NR  N  S  S  W  W  E  H  H  S  S  H  H  E  E
A  A  N  N  N  S  W  W  E  H  H  S  S  H  H  E  E
AAAAA  N  N  N  SSS  W  W  EEEE  HRRR  SSS  H  H  H  H  EEEE  EEEE
A  A  N  N  N  S  W  W  W  E  R  R  S  S  H  H  E  E
A  A  N  NN  S  S  W  W  W  E  R  R  S  S  H  H  E  E
A  A  N  N  SSS  W  W  EEEEE  H  H  SSS  H  H  EEEEE  EEEEE

```

THIS IS THE ANSWER KEY TO THE PROBLEMS GENERATED BY PROBGEN..., DATE: 14-JAN-79

HEADER: PROBGEN II DEMONSTRATION

FORM # PROBLEM # ANSWERS

1	1	1.146354	1.317382	1.753844	12.24254	12.84317
	2	D	-128			
2	1	0.9529921	1.254146	2.097902	12.00812	12.66397
	2	C	-103			
3	1	1.233114	11.71841	12.32340	12.49852	12.54724
	2	A	338			
4	1	2.387020	12.39640	12.63923	12.78470	12.78576
	2	B	1190			
5	1	12.92162	12.90219	12.92145	12.96719	12.96839
	2	B	-110			
6	1	12.76706	12.79364	12.84388	12.93410	13.02191
	2	D	267			
7	1	0.9112884	12.75724	12.81739	12.81871	12.91948
	2	A	-33			
8	1	1.598699	1.996018	12.90312	12.92004	12.92235
	2	B	-251			
9	1	12.57540	12.73459	12.76216	12.82729	12.83902
	2	C	-13			

Figure 2:
Answer Key Generated by PROBGEN II

COMPUTER ACCESS TO INFORMATION:
THE ACADEMIC LIBRARY OF TOMORROW

Robert K. Korfhage
Southern Methodist University

and

Christine L. Borgman
Chilton Corporation

Rapidly escalating costs of information are forcing libraries into a crisis situation. Development of a computer information network based on the academic libraries alleviates this situation and also provides a more completely personalized information service to faculty and students. A key to the success of this information system is the willingness of librarians to be true information professionals.

History has been marked by increased understanding and mastery of our environment, in three great surges. Mankind's first need was to control the physical materials of his world. Increasing sophistication in the use of these materials - first wood, stone, and the easily smelted metals, then metal alloys and the early plastics, and recently an increasing dominance of synthetics - has been well documented. This surge crested with the industrial revolution.

The industrial revolution was made possible by the upwelling of the second surge - control over energy sources, such as coal, oil, and gas. This surge continued with the development of electricity, atomic and nuclear energy, and the first serious work on solar and other energy sources. Perhaps we are too close, but the second surge appears to be creating just as the third one is coming into dominance.

It is this third surge - control and communication of information - that concerns us. There are four great milestones in the history of information and communication: writing, the alphabet, moveable type, and the computer. Writing allowed man to record information accurately for his own and future generations. The alphabet materially improved the ability to retain information, providing a systematic, efficient way to record facts and ideas. Moveable type made information generally available, through the rapid production of many copies of any document. The computer can now store vast quantities of information and disseminate this information on request, with great speed and very low cost. We have entered the era of the information society.

For centuries, libraries have been repositories of recorded information, passively awaiting the user. School and academic libraries have traditionally formed a significant, but unstructured, part of a learning environment. Students and faculty do

not allocate specific hours and syllabi to libraries, as they do to courses, but rather use the library as a pool of knowledge upon which they may draw as needed. Libraries are expected to contain a great diversity of information - on art and science, health, amusement, current events, history, and so forth. Keeping this vast variety of information current has become impossible, as the rate of production of documents grows. It has been pointed out by Lancaster, for example, that not only is the amount of material being published in any given scientific journal increasing exponentially, but also the number of journals is increasing exponentially. Combined with the rapidly increasing costs of book and journal production, library space, and library operations, there can be only one conclusion: the traditional academic (or public) library cannot survive for long.

Libraries are collecting and storing huge volumes of paper in a relatively inaccessible way. The incorporation of microfilm and other media into the system only exacerbates the problem: the traditional modes of storage, indexing and access are not well-matched to the new media. Libraries must move toward supporting information in a more fluid and flexible form which can be utilized through existing and impending technology. Technology is available to make information access rapid and efficient. If this technology is not already cheaper than traditional methods, it will be within a very few years.

For a number of years, computers have been touted as the answer to the libraries' problems: computerize the library and its holdings, and all will be well. Yet the moves in this direction have been small and largely ineffectual, for many reasons. The cost of installing and maintaining a computer system has been high. The cost and effort of converting present library holdings to computer is staggering. Library space must be redesigned to utilize the computer, and

staff must be retrained - both expensive. There is the problem of public acceptance of a computerized library, and the concomitant problem of training the library patrons in the new methods of operation. And of course there is plain old inertia - we've always done it this way; it has worked up to now: why change? We shall change because we must!

It seems to us that during the past year or two a significant change has occurred in computing, greatly reducing the impact of most of the objections to a computerized library. We refer, of course, to the development of microprocessors, and the changes that have resulted in computing - in cost, in speed of access, in capability, and in general acceptance. The pocket calculator that two years ago cost \$100 or more now costs \$5: it is a "throwaway." A similar change has taken place in the price of computers: computing power now costs a tenth of what it did just two years ago, or even less. The capabilities of computers are changing at a similar rate: the speed of computers has increased by a factor of one hundred or more; the density of computer storage has similarly increased in just a year or two. And these changes continue. Dare we predict? Within five years a computer system capable of handling a moderate size library will cost \$2,000 or less, assuming one terminal. (The cost of terminals has not been decreasing nearly as fast. Each additional terminal still costs several hundred dollars.) This computer will be able to handle all library functions, from the routine business operations, through ordering and cataloging, to serving as the patron access point to the library collection, to sending out overdue notices. However, as we shall see, in a fully developed library information system, the importance of routine operations will diminish, and patron access with professional assistance will become a dominant mode of the system.

For a computer to be really useful in a library, both the librarians and the patrons must be able to operate it. Two factors indicate that this will be only a minor problem. First, while computer languages have been quite arcane, and not at all suited to use by untrained people, this situation is gradually changing. New languages are more specifically user-oriented, permitting the user to manipulate the computer system in terms that are natural to his or her background. In addition, many of the standard operations, such as various business and accounting functions, are being built directly into the computer system. It has been predicted that the first computer software store - selling programs of all types - will open by 1980. We venture to suggest that many of these programs will not be available in the traditional forms of listings, cards, or tapes, but will be built into "ROMs" - read only memories, that you can plug directly into your computer. One manufacturer is already offering such program ROMs for the top model in its pocket calculator line.

The second factor that indicates that computer operation will be a minor problem is the fact that today's students, the library patrons of now and the future, are growing up in an environment that impels ready acceptance of the computer and its associated equipment as an information tool. Television, long decried as turning children away from books, is nevertheless a powerful medium of information. It is doubtful that any child growing up in this environment would object seriously to receiving information via a video terminal. The TV games that have

come on the market within the past two years have added the element of activity to television: no longer is the child a passive recipient of what is shown on the tube; he can and does control that information. Pocket calculators, home computers, computer-assisted instruction in the schools: all point to the fact that the child growing up today will be perfectly at home with a computerized information system. He will interact with the system freely, making requests to the system, and receiving back the information that he wishes. If only the system is there!

John Kemeny recently questioned whether we are raising a nation of computer illiterates. Our fear is not that. We see the younger generation as at least semi-literate - able to use a computer information system, although perhaps not fully understanding what it is doing. But we do see a great deal of computer illiteracy among adults; and we do fear that this illiteracy will have a detrimental impact on the students of today and tomorrow, through a reluctance to provide these students - and to anyone who can use it - a computerized information system built upon our current library system. We propose that the time has come to move strongly into computer-based library and information systems; that such systems should provide direct and remote access by the patrons; that realistic move must be made toward covering the cost of any information system; and that the school and academic libraries provide the proper ground for the development of such systems.

We wish to place before you a seemingly utopian library system, to argue that such a system is not as unrealistic as it might first seem, and to explore the manner in which an academic library might move toward such a system.

Consider this fact: The only reason that a library - or a group of libraries - needs multiple copies of a book is so that several patrons may use it at once. Faced with the same problem for computer programs, computists developed a reasonable solution: provide one copy of program, written in such a way that any user can access it at any time, regardless of other current users. Apply this solution to libraries: why not have, for each book, journal, or paper, one copy that is accessible to all users? One copy, that is, in the country, or even in the world. Putting aside the non-technical problems this would raise, let us consider the needed technology, and the impact on the academic library.

The underlying technology is a computer network, such as already exist commercially. A document is written and edited at a network node using the computer text editing system. When the document is ready, the information retrieval system is notified of its existence, and the proper indexing and location information is established. The user (i.e., library patron) may then query the network from his home terminal and retrieve the document, should it match his needs. All of this is technically feasible today; it will be economically feasible within a year or two.

How does this impact the academic library? In its "museum" aspects - collections of rare books - the library will be little affected. However, as an information resource, the library will be greatly transformed. First, physical growth will virtually cease. Much of the information that the students and faculty access will not be stored locally, and that which is locally held will be in a highly compact, computer-accessible form. In fact, as currently held

60 information ages, the useful part will be converted to computer form and the obsolete part will be discarded, so that the physical size of the library may actually decrease.

Second, the library will function primarily as a node in an information network. Even today, no academic library can stand alone. Interlibrary loan is vital to the life of the largest and the smallest. As we move to a "one document - one copy" library system, ILL will become ILA - interlibrary access, and assure a dominant role in library activities. The library's own "collection" will become specialized - works written by local authors, or works in some topic area.

Third, the library staff will become much more computer-adept, and will assume an increasingly active role in information retrieval. Recent research at Syracuse University suggests that even trained professionals interpret questions quite differently, and that the use of several professionals to query an information system may dramatically improve recall without a loss of precision. The librarian will no longer catalog and shelf books: there will be no book shelves, and document cataloging will be largely automatic. The librarian will be able to provide truly professional assistance to the patron in his quest for information.

Fourth, as this indicates, many of the routine tasks of today's libraries will be automatically done by the system, or will be unnecessary. Ordering and cataloging will disappear, as the library has access to everything in the system, and new entries are automatically cataloged. Overdue notices will no longer be needed, as information is used, not borrowed.

Fifth, librarians will need to become much more sensitive to the real information needs of students and faculty. Very largely, a person's access to information is limited by his ability to articulate a question. Questions rarely are as neatly packaged as documents are; and currently they must be force-fitted into the library access system. In a broad information network the fit will be less rigid - but this implies that the user must better articulate the question to avoid a flood of responses. The librarians will assume the role of "teacher of information access" - training students to better understand their real information needs, to formulate more precise questions, and to properly interpret the results of a search. The librarian will become a professional in the highest sense of the word.

We have suggested that the patron will access the information system from a home terminal. Why should this access be through the library? The driving forces will be economics, convenience, and the presence of professional assistance - exactly the same reasons that bring a patron into a library rather than into a bookstore.

There are non-technical problems associated with the implementation of such a system - problems of an economic, legal, or political nature. We feel that these problems are not materially different from those being faced today by library and other information systems.

The primary economic problem is that of supporting a widespread information system. There is a long-standing tradition of "free" library services - that is, that library costs are largely covered by general tuition revenue, tax money, and other sources, with essentially no direct charge to the patron. Escalating library costs

cause us to doubt the feasibility of this funding approach, particularly for a traditional paper-bound library. Whether the economics of a computer information network will permit such "free" services remains to be seen. Certainly one must consider the other side of this coin also - adequate recompense to authors for the use, rather than the sale, of their writings.

Legal problems center largely around copyright, but these are problems that exist whether or not we have a computer information network. We contend that these problems arise from a lack of understanding of information as a commodity, and must be solved independently of the form of a library system.

Political problems, such as the access to governmental or sensitive information, are also a present problem, independent of the library's form. Within an academic environment these presumably cause less difficulty than they would in a public library system. The primary impact of a computer information network here would be on maintenance of access control, so that patrons could not use one network system to access another (forbidden) one. This, however, is a problem that commercial computer networks are currently facing and solving, and is not unique to libraries.

The computer may amplify problems - just as it amplifies benefits - but it will not create new ones.

Summary

We challenge the academic community: get off your archives and lead the way! Install a computer system; build up a database; train your faculty and students. Start with a topic of significance to your school and become the authority. Open your database to computer access by student/faculty/public. Learn to be an information professional. Develop a real information system - not just a computerized catalog or some automated business functions. Begin with faculty authors who are willing to use the computer text editor and leave their work there for access. Use this work as a base for your information system. Encourage faculty and students to help you build the system by identifying documents to be added to the database. Each nuclear collection thus started - and you can control the number - will grow as its presence becomes known, and as other patrons, both on-campus and off, begin to use it.

Become aware of similar efforts at other schools, and work with others to develop compatible systems that can build into one large network. Encourage the expansion of the information networks into other topic areas: there is more than enough information to go around.

Most importantly, get started! The need is apparent, and by acting now your library, your school will play a significant role in the development of this coming library information network.

References

1. C. L. Borgman and R. R. Korfhage. The public library interface to personal computer systems. ASIS Annaul Meeting, New York, November 1978.
2. M. Koll, J. McGill, T. Noreault. Individual differences in online searching. 1979 ACM Computer Science Conference, Dayton, Ohio, February 1979.
3. R. Korfhage. The impact of personal computers on library information systems. SIGIR Forum, Spring 1978.
4. F. Wilfred Lancaster, Whither libraries? or, wither libraries. College and Research Libraries, 39 5, September 1978, 345-357.

COMPUTER ASSISTED TEST
CONSTRUCTION AT THE
UNIVERSITY OF GEORGIA

John S. Edwards
James E. Eisele

University of Georgia

The University System of Georgia Test Construction System provides large files of test items, an interactive retrieval program, test and item analysis, and personnel trained in test development for the faculty. The content of the test item files is suitable for secondary and higher education students. The computer system consists of vendor utilities and locally written software and runs on a CYBER 70/74. The use of TCS for preparing student self-study guides and classroom examinations has increased during the two years of its use. Additional item files are acquired regularly and the faculty continues to add items to existing files and to create new files.

INTRODUCTION

This report describes a computer-assisted test construction system in the University System of Georgia Computer Network (USCN).

The USCN is composed of thirty-two institutions of higher education which share the computing resources of the Central Site. Currently the Network utilizes a Control Data CYBER 70 Model 74. The Division of Educational Support functions as a unit of the USCN. This Division is composed of personnel who have a background in, and understanding of, educational innovation supported by personnel with technical competency in computer applications.

The two major functions of the Division are (1) to select, evaluate, and acquire existing courseware and software for educational applications, and (2) to provide leadership in creating both courseware and software. The Georgia University System Test Construction System was developed as an application of this second function.

A search of available test construction computer systems revealed that either the programming languages were incompatible with USCN hardware or the design did not satisfy the requirements of the Network faculty. Consequently, a system was developed using vendor utilities combined with local design and programming. The locally created modules are written in FORTRAN and run on the CYBER 70/74. The system incorporates

elements of interactive and batch processing.

PURPOSE

The TCS consists of more than computer programs. The primary purpose is to provide a central source for the following services:

1. Developing tests for classroom, department, school, or system use
2. Constructing exercises, examinations, study guides, etc. via computer program which searches large data files
3. Scoring, analyzing, and interpreting tests used throughout the system
4. Maintaining files of calibrated test items for use by the students and faculty of the system
5. Providing personnel trained in test development and computer applications for consulting with the faculty.

These services are provided by implementing the following six functions:

1. Information Dissemination - The faculty is informed of new acquisitions, workshops, seminars, etc., by means of a user bulletin which is published periodically. In addition, a series of descriptive articles is carried in the official news organ of the USCN. The workshops provide an overview of the TCS services and acquaint the faculty with the means for accessing the system

and with the content of the test item files. These workshops are held monthly on the Central Site campus and by request at the remote sites.

2. Test Item Acquisition - The TCS has an active program of test item acquisition. The files currently contain over 50,000 items in twenty different subject areas. (See Figure 1) These files are continuously updated and increased by the faculty. The Division staff constantly monitor the literature for news of item bank development in order to enhance the holdings.

3. Test Construction - A conversational mode interactive program for searching the item files and printing the test items was designed and programmed by the Division. This program prompts the user to provide information necessary to select appropriate items.

4. Test Scoring - Currently there are two scoring and analysis programs available to the faculty. One, named STANAN, is designed expressly for the classroom test and for faculty not familiar with test and item analysis statistics. The most valuable feature is the inclusion of explanations of the meaning of and guidelines for interpreting each statistical index. ANLITH, on the other hand, provides more sophisticated statistical analyses suitable for the psychometrician.

5. File Maintenance - The Division of Educational Support has the responsibility for storing and maintaining the test item files. The computer storage of the files is provided as a service to the University System thereby eliminating the redundancy and expense associated with having the extremely large files stored in each user library.

6. Test Consulting - The Division staff provides expertise in test development which is available on a consultant basis to the faculty. This service includes interpreting statistical analyses, test development (preparing objectives, writing test items, etc.), indexing test items for subsequent retrieval via the search program, as well as guidance in creating item files for personal use.

The interactions of faculty, Division staff, computer programs, and item files are diagramed in the flowchart in Figure 2. The processes implemented by the faculty are outlined with broken lines; those of the Division are outlined with a solid line. The basic philosophy underlying the use of TCS is that it be made accessible to the novice computer user with as little interface of the Division staff as possible. The user manuals are prepared in such a manner that we expect, after minimum training of about fifteen minutes, the user can retrieve and print appropriate test items.

As presently constituted, the Service provides an interactive computer program for selecting test items, test and item analysis programs, large files of multiple-choice and true-false items, and facilities for editing and updating the files. The faculty is encouraged to use the item files as a basis for developing their own instruction/testing program. Other files will be added as they become available. The faculty is also encouraged to construct files of test items in subject areas which are not currently available. Additionally, the present files may be augmented by including items generated locally.

Selecting Items. Two files are required for this process. One file, named the datafile, contains the test items. The other, the logfile, contains the index to the test items in the datafile. The logfile contains a series of code numbers with descriptions and the quantity of items indexed under each code number. An example of a logfile for Psychology is shown in Figure 3.

CATEGORY NUMBER	NUMBER AVAILABLE	CATEGORY NAME
70000	1	The Normal and Abnormal Personality
71100	22	Personality

FIGURE 3. EXAMPLE OF A LOGFILE

The column labelled "Category Number" contains a five digit code number which represents a content area. The content area is given a brief description in the column labelled "Category Name." In the example, Category 70000 has one test item indexed as indicated in the column labelled "Number Available" and Category 71100 has twenty-two test items.

Items are selected from computer files by means of an interactive retrieval program. This program prompts the user through a series of requests for data. The following discussion presents the steps the user passes through during item selection.

Step 1: Datafile Request. After the logon procedure the user issues two commands which name the programs in the local library and execute the programs. The TCS prompts the user for the name of the datafile and corresponding logfile.

Step 2: Content Category Selection. The main access point to the test item is the content category. The content category codes for each datafile are published separately or may be obtained during the item selection process.

Step 3: Item Type and Quantity Selection. After entering the Content Category selection the user is requested to enter the item type code and quantity of items

<u>SUBJECT AREA</u>	<u>DATA FILE NAME</u>	<u>NUMBER OF ITEMS</u>
Biosciences	BIOLOGY	2182
	FOOD	556
	NUTRITION	217
Mathematics	STAT	206
	MATH	3386
Liberal Arts	HISTORY	6086
	FRENCH	120
Physical Sciences	PHYSICS	1034
	ASTRO	475
Social Sciences	INDPSY	1028
	PSCH	8000
Chemistry	CHEM	6564
Business and Finance	BUSINFO	2040
	ECON	2756
	FINMAN	1130
	ACCOUNT	6475
	MARKET	4000
Computer Science	DPDATA	1689
	TESTM	2000
Education	TEACH	173
	COUNSEL	1281

FIGURE 1. TCS TEST ITEM FILES

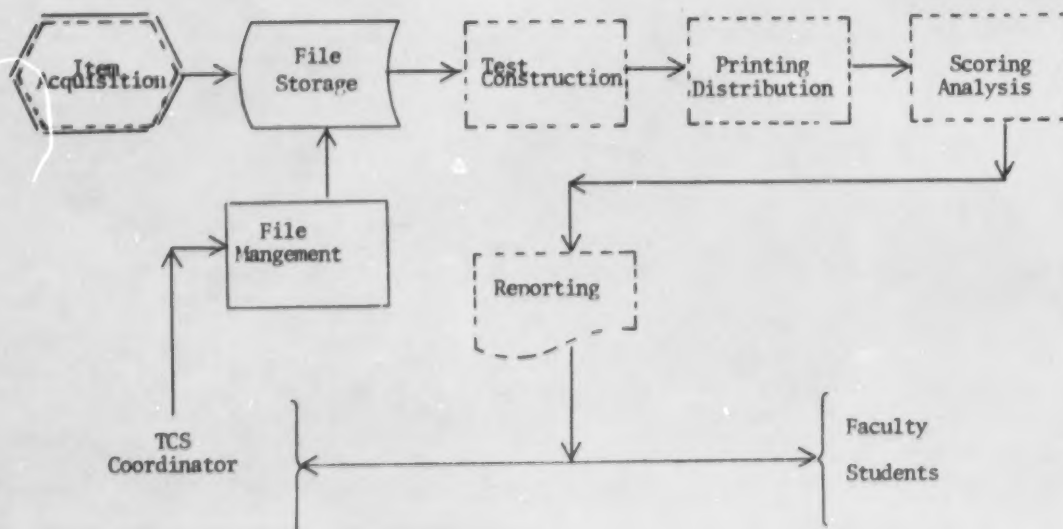


Figure 2. Test Construction System Interactions

desired. The TCS can store and retrieve true-false multiple-choice, short-answer, and essay item types.

Step 4: Cognitive Level - Difficulty Level Selection. The choice of Cognitive Level and Difficulty Level is optional. The computer will provide an equal distribution of these levels within the quantity of items requested. Usually those interested in test development from a psychometric standpoint will opt to choose cognitive and difficulty levels. Step 5: Item Review. The TCS provides an opportunity for review of each item which has been selected. During this review, the user may delete any items found to be unsatisfactory.

After item review the user may choose to select another logfile and continue selecting items, end the item selection process and print the test, or end the session without printing the items.

Step 6: Item Printing. In order to ensure proper identification and delivery of item print, the TCS requests an item booklet identification, instructor's name, test name, date, school name and mailing address.

After entering the requested information, the user is given the choice of printing the items of three different locations. They may be printed at the user's data terminal, the local computer room, or at the network central site.

Test and Item Analysis. The standard test and item analysis (STANAN) is available both as a function of TCS and independently. That is, any set of item response data regardless of source can be analyzed with STANAN.

STANAN provides test and item analysis for multiple-choice and True-False type items. A maximum of five options per item is permitted.

The descriptive statistics provided for tests include: Group size, Mean, Standard Deviation, Reliability Estimate, and Standard Error of Measurement.

Student scores are reported as raw scores, T-Scores, percentiles, and percent correct. A listing of scores by student name and by a nine-digit identification number is also provided. The item analysis data which is reported includes: Percentage of responses for each option including omitted responses, Frequency of responses for each option, Difficulty index, and Discrimination Index.

The most attractive feature of STANAN is not in the actual Statistics reported as these are usual and to be found in every program of a similar nature. The popularity is due to the manner in which the data is reported. Each statistic is explained and guidelines for interpretation and application to improvement of tests and items are printed along with the data. The majority of the faculty is not accustomed

to reading computer print-outs and certainly not comfortable with statistical analyses of tests and items. Therefore, the suggestions made for using the statistics are most useful.

SUMMARY

The TCS has provided a useful service to the USCN. Though no formal monitoring of use has been conducted, informal feedback has been quite favorable. The quantity of inquiries for user manuals and consultation has increased rapidly in the two years since the service was introduced. Additionally, test item files have been built in the disciplines of Pharmacology, Poultry Science, Veterinary Medicine, Music, Health Sciences, and Physical Education.

Projections for improving the service include providing data entry programs for building item files and entering response data for analysis, an interactive test program for on-line student testing, and continuing enhancement of the test item files.

THE COMPUTER ATTITUDE CONTINUUM
(CAC): SORTING THE CRITICS
AND THE APPRECIATORS

Walter M. Mathews
The University of Mississippi

Abraham W. Wolf
The University of Mississippi

With the computer becoming ubiquitous it is important to develop empirical means whereby attitudes toward computers in particular and technology in general may be assessed. The Computer Attitude Continuum has the desirable psychometric properties to be an effective means of evaluating positive and negative attitudes towards computers.

The CAC was administered to several undergraduate classes that were composed of computer users and non-users. An oblique rotation of two factors indicated that the scales were factorally discriminable. Split-half reliability and internal consistency of the two scales were quite good. Furthermore, the instrument performed better than chance at distinguishing between groups in a discriminant analysis. It is suggested that the CAC is an effective means of assessing attitudes towards computers and technology. Those employing computer technology in educational settings will find the instrument especially useful in assessing attitudes.

INTRODUCTION

Attitudes are evaluative beliefs. They are likes and dislikes. They are our affinities for and our aversions to situations, objects, persons, groups, or any other identifiable aspects of our environment. Our likes and dislikes have roots in our cognition, in our emotions, in our behavior, and in the social influences upon us (Bem, 1970).

Twenty-eight years after it first was made commercially available, the computer appears nearly everywhere in our culture: in our offices, stores, schools, automobiles and homes. As IBM reminds us: "It was to have been the Nuclear Age. It became the Computer Age."

Our world is dependent upon computers, with little chance of trend reversal. The Industrial Revolution provided technology that supplemented the muscles of humankind--and brought with it enormous social problems.

The Computer Age is providing technology that supplements the mind of humankind--and it too brings social problems . . . problems of a magnitude previously incomprehensible: problems concerning individuality, privacy, power and automation.

In the beginning of the Computer Age, most human attitudes toward this new technology were irrational: "It's an

amusing toy," "A mechanical man," "A superhuman brain," "Godlike," "A tool of the devil," et cetera.

The pervasiveness of computers today demands that we study societal attitudes toward it for at least two reasons: to better understand and correct the fallacious and irrational attitudes toward this integral component of modern life, and to better understand the rational attitudes against computers and their uses so that the individual and society may be protected.

The purpose of this study is to devise an instrument to measure the positive and negative attitudes individuals have toward computers and associated technology.

METHOD

SUBJECTS

The respondents were undergraduate students enrolled in educational psychology, radio and television, and computer science courses. Students were sampled in this manner in order to obtain a diversity of respondents so that the stability of the results across various populations could be examined.

A total of 216 students from seven classes completed the CAC. Of these, three protocols were deleted since the data were not complete. The sample was

primarily composed of freshmen, sophmores, and juniors; the sex distribution was equal.

ADMINISTRATION

The CAC containing 50 items plus 5 items of demographic information was administered to seven classes. The subjects were requested to respond to the CAC items on a Likert scale from "strongly agree" to "strongly disagree." They were advised by the test administrator that they were participating in a data collection for research on computers.

RESULTS

The 50 items of the CAC were initially factor analyzed by a principal components method (unities in the diagonal) using the eigenvalue greater than 1 criteria to extract factors for rotation (Dixon, 1977). Twelve factors were extracted in this manner. To determine the non-trivial factors, the "break-in-the-roots" method, or scree test (Gorsuch, 1974) was employed. This suggested three factors for rotation.

An examination of the oblique rotation of three factors indicated that the third factor did not greatly add to the variance accounted for and confused interpretation. Pilot research indicated a two factor solution and this was also indicated with the present data as being more parsimonious.

The results of the oblique rotation of two factors are presented in Table 1. Both factors account for 31.69% of the total variance (Factor I: 16.69%; Factor II: 14.73%). A variable was assumed to have a salient loading, i.e. to be related to a factor if: (1) the factor loading was greater than .30 and (2) if an absolute value of .30 or greater was obtained on one and only one factor. Otherwise the variable was labeled as uninterpretable or marginally interpretable.

The items of the CAC are worded such that their content reflect a positive or negative attitude toward computers and their effects. An example of the former is Item 17, "Computers are improving our lives;" of the latter, Item 16, "Computers have too much control over people's lives." It was expected that items with common content would load on the same factor. Results confirmed this expectation.

To aid in the interpretation of results, the items and their factor loadings are organized in Table 1 according to the content wording of the item. For

the purpose of inclusiveness the groupings have been termed Content-Negative and Content-Positive. With the exception of several items all of the Content-Negative items load either positively on Factor I or negatively on Factor II; this is also the case for the Content-Positive grouping leading positively on Factor II or negatively on Factor I. This grouping of items with factors suggests that the two "scales" are factorially discriminable.

The correlation between the two factors is -0.36. This moderately negative correlation suggests that while there is some relationship between the factors, this relationship is negative. It would appear that similar but opposing dimensions are being gauged by the two factors. This is consistent with the aim of the investigators to determine dimensions of support and criticism toward computers.

Reliability coefficients were calculated for the 33 items of the Content-Negative scale and the 17 items of the Content-Positive scale. Split-half and internal consistency criteria were employed.

The correlation between split-halves of the Content-Negative scale was 0.82. The Spearman-Brown coefficient which predicts the increase in reliability from a hypothesized increase in the number of variables was 0.90. Coefficient Alpha, a measure of internal consistency or homogeneity of the items, was 0.92.

The correlation between split-halves of the Content-Positive scale was 0.75 which increased to 0.86 when the Spearman Brown coefficient was used. Coefficient Alpha was 0.87.

The question was asked as to how well the two scales could discriminate among the groups that composed the sample. Two groups were formed from the 7 classes: Group 1, the non-users of computers, was composed of students in educational psychology and radio and television classes; Group 2, users of computers, was composed of students in computer science classes. A discriminant analysis was employed using the two groups as the independent variables and the Positive and Negative scales as the dependent variables.

One discriminant function was derived ($p < .005$). The discriminant function coefficients were as follows for the scales: Negative: 0.809; Positive: -0.294. Group centroids, or means, were Group 1 (Humanities): -0.276; Group 2 (Computer Science): 0.176. This suggests that a positive attitude is associated with users and a negative attitude with non-users.

Table 1
Principle Components/Oblique Rotation of CAC
Items by Content Grouping

Content-Negative Items			Content-Positive Items		
Item	Factor I	Factor II	Item	Factor I	Factor II
1*	.25	-.07	6	.12	.46
2	.49	-.02	7	.09	.64
3	.63	.01	8	-.35	.25
4	.43	.10	9**	.20	.17
5	.55	.21	12	-.10	.64
10	.05	-.39	17	-.15	.64
11	.64	.01	19	-.15	.59
13	.19	-.35	21	.30	.65
14**	.21	-.21	27	-.18	.43
15	.46	-.17	28	.00	.60
16	.71	-.09	30	.05	.62
18	.60	.06	32	.08	.63
20	.42	-.23	34	-.03	.54
22	.52	-.02	36	.01	.68
23	.68	.27	37	-.11	.57
24	.26	-.47	40	-.06	.53
25	.64	-.00	48	-.13	.47
26**	.38	-.41			
29	.50	.05			
31**	.32	-.42			
33	.55	-.09			
35	.41	-.27			
38**	.38	-.38			
39	.40	-.00			
41	.42	-.14			
42	.76	.18			
43	.37	-.18			
44	.38	-.17			
45	.48	-.16			
46	.54	-.06			
47	.65	.02			
49	.64	-.14			
50	.50	-.26			

*Marginally interpretable item

**Uninterpretable item

A classification matrix predicting group membership from the discriminant function coefficients indicated that 61.97% of the cases were correctly classified. That is, the instrument was approximately 12% better than chance at predicting group membership.

DISCUSSION

The easy discrimination of the items as being positive or negative by their content alone suggests that the CAC possesses a high degree of face validity. This grouping of items by content was confirmed by the factor analysis where items having common content loaded on the same factor.

Such terms as positive and negative, though, lack specificity. It is suggested that Factor I, the negative factor, be termed Critical while Factor II, the positive factor, be termed Appreciative. Such labels would more adequately reflect the notion of a posture or stance toward the use of computers and their effects on the quality of life.

There is a problem with the utilization of the instrument in its present form. The two dimensions of Critical and Appreciative are not balanced in terms of the number of items that constitute the two factors. This would make for major problems in scoring. The majority of the items are worded to be critical of computers. Such an imbalance can hardly lend the instrument--at present at least--to being called a continuum. Clearly, further work is indicated.

Yet, there is good evidence to suggest that the two factors constitute a continuum. The two factors do show a moderately negative correlation indicating that they fathom a similar phenomenon--i.e. attitude towards computers and technology--but observe facets of that phenomenon. On this basis we feel justified in calling the instrument a continuum.

To provide a balance to the continuum, though, there should be an equal number of items to establish equal but opposite extremes. Further work on the instrument will attempt to provide that symmetry.

With the introduction of the computer in a variety of environments many of which may harbor less than positive feelings toward the new arrival, the CAC may prove to be a useful instrument for assessing attitudes. Teachers and administrators could be evaluated as to their receptivity toward the introduction of educational applications of the

computer through administration of the CAC before and after implementation. Students using computers in their studies could be similarly evaluated. By being sensitive to the attitudes and feelings of those who will use computers, by anticipating the common intimidation and hostility of those not initiated into its "mysteries," an environment can be created for its use by appropriate information and education. The CAC may be the means whereby such attitudes may be assessed.

REFERENCES

- Bem, Daryl. Beliefs, Attitudes, and Human Affairs. Belmont, CA: Brooks/Cole, 1970, p. 14.
- Dixon, W. J. BMDP-77: Biomedical Computer Programs, P-Series. Berkeley; University of California Press, 1977, pp. 656-684.
- Gorsuch, R. L. Factor Analysis. Philadelphia: W. B. Saunders, 1974, pp. 152-156.

COMPUTER-BASED INSTRUCTION IN MEDICAL EDUCATION

William E. Sorlie

&

Diane L. Essex

University of Illinois,
Urbana-Champaign

The School of Basic Medical Sciences, Urbana-Champaign (SBMS-UC), University of Illinois, College of Medicine, sought and received monies from the federal government (HEW) to develop a computer-based medical sciences curriculum on the PLATO IV computer system. The major goal of this developmental effort was to produce a substantial body of computer-based material which would become a functional component in the ongoing, self-paced instructional program at SBMS-UC. The computer-based program was to also utilize the technological advantages and flexibilities inherent in the PLATO IV computer system. A majority of students felt the computer lessons helped them to learn the basic medical science content. However, in almost all analysis of testing data, no significant differences in user and nonuser performance could be found.

Introduction

The teaching of basic sciences to medical students has until the past few years been primarily didactic in nature, relying mainly on lectures, laboratories and textbooks. However, educational technology, specifically computer-assisted instruction, has made an impact in the development of new and in the revision of established medical school curricula. One medical school in particular has been using many components of educational technology in the delivery of its educational program.

During the period (July, 1973 - July, 1977), efforts were undertaken at the School of Basic Medical Sciences, Urbana-Champaign (SBMS-UC), University of Illinois, College of Medicine, to develop a computer-based curriculum in the basic medical sciences by using the PLATO IV computer system. These development efforts were supported, in part, by a Department of Health, Education and Welfare, Bureau of Health Manpower contract (No. 1-PE-34-68). The contract entitled, "PLATO IV Applications in the Basic Medical Science," was for four years. Pursuant to this contract, a comprehensive evaluation of the project was conducted. The Office of Curriculum and Evaluation (OCE), School of Basic Medical Sciences-UC, was contracted in July of 1975 to develop and conduct this evaluation.

Background

The School of Basic Medical Sciences at

Urbana-Champaign has been involved in an innovative educational approach to the preclinical year(s) of medical education. When SBMS-UC opened in the Fall of 1971, basic science faculty and local practicing physicians prepared a basic medical science curriculum to provide medical students with the skills and knowledge necessary to allow entry into a clinical training program after 10 months of study. The curriculum contained 12 basic medical science disciplines and was organized into a five-volume document. The curriculum contained over 400 self-instructional units and incorporated over 4,000 specific learning objectives. Each curriculum unit included all necessary components to allow the student to proceed at his/her own rate of progress and in a manner consistent with his/her own learning style(1).

SBMS-UC PLATO IV Proposal

To complement and add support to the existing SBMS-UC educational program, the School, in 1973, sought and received monies from HEW to develop and implement the Computer-Based Education Project (CBEP). Some of the more important features of the initial proposal and ensuing four-year HEW contract were:

1. A core curriculum of approximately 300 PLATO IV computer units was to be developed.
2. The materials developed by the CBEP were to cover the same general scope as the SBMS-UC curriculum.

3. Developed computer lessons were to include learning objectives, self-testing and evaluation devices.
4. A modular format was to be developed that would allow for multiple entry and exit points within each module that would permit students with different training levels equal access to the same materials.

Project Faculty

Staff for the project included one M.D. full-time director, five Ph.D. content authors, a small number of graduate student programmers and associated clerical staff. Most project authors were recently degreed and had neither a great deal of teaching experience in medical schools nor experience in computer-based instruction. In anticipation of the intensive efforts required to adapt the new system to health sciences, project staff were full time with no research assignments. All project Ph.D. faculty were accorded adjunct faculty status in the school; however, none were given any formal teaching responsibilities. It was intended that the project staff and the School faculty would work jointly in the definition, development and review of all computer lessons⁽²⁾.

Methodology

The evaluation was designed to assess the progress of the project relative to the goals outlined in the contract (summative evaluation), provide feedback to the project and School personnel, as well as to the funding agent (summative and formative evaluation), and portray the project as it had evolved since its inception.

As a first step in the evaluation, the objectives of the contract were outlined. Three major objectives were identified: development of course teaching material, implementation of the course material and effectiveness of the course materials. These objectives were then delineated into small (sub) objectives which could be measured more easily. For each (sub) objective, the evaluation questions were identified, data sources specified and analysis strategies outlined. Data were gathered from varied sources. Interviews were conducted with various participants in the program. Opinions about the courseware were also obtained from medical students via questionnaire.

Lesson Effectiveness

Student usage of PLATO lessons and subsequent performance of students on relevant portions of School and College examinations were analyzed. Means and

standard deviations were calculated. A t-test was used to examine performance of "users" and "nonusers." Based on the twelve analyses conducted, only one revealed significant performance differences favoring the user group. In the remaining analyses, it was observed that, although some performance benefits accrued to users of PLATO lessons, differences between users and nonusers were not significant. In addition, no performance benefits were found to exist for students who had used PLATO lessons related to subtests on the end-of-year college certifying examinations⁽³⁾.

Student Opinions of Lessons

As part of the total evaluation of the approximately 150 developed PLATO IV basic science lessons, a questionnaire was sent to the medical students asking their opinions of the lessons. The questionnaire consisted of eight questions; six of the eight questions contained several subparts. A Likert-type responsescale was utilized. Responses to some questions were on an agree-disagree continuum; responses to other questions dealt with the helpfulness of lessons and the frequency with which certain learning resources were employed.

Only students who had used the lessons could respond to many of the questions. Students who indicated they had never used the lessons were sequenced through the questionnaire in such a way that they responded to general questions requiring no exposure to the lessons. Of the 97 students enrolled in the Urbana program, 81 returned questionnaires. Of these 81 respondents, 65 indicated they had used the basic science lessons; 15 indicated they had not used any lessons.

Overall Opinion and Summary

Students were asked their overall opinion of PLATO as a learning tool (i.e., "the PLATO basic science lessons helped me to learn the basic sciences" and "PLATO is not a useful learning tool"). Sixty-two percent of those who had used the lessons agreed that they helped them learn basic science concepts. Almost 90% of the students disagreed with the second statement. They thought PLATO was a useful learning tool. Further, 92% of the individuals indicated that the lessons support the School's philosophy of self-paced learning. Three-fourths of the students said they had used the PLATO basic science lessons primarily for instructional, as opposed to recreational, purposes. Considerably more than half of the lesson users (85.5%) felt that the basic science lessons were well-written. These students indicated that lesson content

72 was neither too elementary (85.7%) nor too difficult (96.9%) to be useful. When asked if their problem-solving skills were improved through the use of the basic science lessons, 39.7% of the students who had used the lessons said they were. When asked if they liked the problem-solving exercises provided, 85% responded affirmatively.

Most (90%) students liked the component of being able to gain access to the lessons through an on-line catalogue. Approximately 92% of the student users liked being able to correspond on-line directly with the author(s) of lessons. They felt that this feature permitted them an active role in lesson improvement and development.

Nearly 70% of the students who used the lessons liked the explicit listing of the educational (learning) objectives of the lesson. About 85% of the student users liked a lesson format in which the key words and concepts of a lesson were identified. Roughly 87% of the students liked being able to approach a lesson in more than one manner, i.e., various paths or branches in a lesson were favored. Nearly all users of the lessons (95%) favored more than a simple "no" judgment for incorrect answers. Brief explanations and/or hints to the correct answer were felt to be more beneficial to learning. Additionally, about three-fourths of the student users liked the use of pre- and post-test questions in lessons. Students wanted to know exactly how well they had done immediately after the instruction.

Review for Examinations

Since the students at the Urbana campus take a number of comprehensive examinations during and at the end of the year, they were asked if they did or would use the basic science lessons to review for these examinations. Ninety-two percent of the students who had used the lessons said they did not use them primarily to review for the examinations given during the year. Further, 56% said they also did not plan to use lessons to review for the examinations at the end of the year.

It was somewhat surprising to find that students do not plan to use the lessons as a review tool, in light of the large number who had expressed that the lessons were helpful in learning the basic sciences. One possible explanation is that students found that PLATO basic science lessons progress too slowly and require more interaction than needed during review. A textbook can be scanned quickly. A list of learning objectives can serve as a checklist, but PLATO lessons designed to teach a

concept might be laborious the second or third time, when the student wants to refresh his/her memory. Another possible explanation is that students have studied from textbooks for the major portion of their academic career and are more confident and comfortable with this medium. A third hypothesis is that the extant body of lessons does not cover in a cohesive fashion and/or adequate manner the material to be addressed on the examinations. This latter conjecture is supported partially by comments from students that additional lessons are needed in several basic science areas. Some students also noted that more lessons were needed to close gaps in existing courseware.

Lessons and Other Learning Resources

When asked if the basic science lessons were good supplements to the curriculum, 91% of the students who had used them said they were. Students were asked if the lessons supplemented regular (non-PLATO) laboratory exercises and demonstrations. These responses were divided about evenly on this question--51% agreed that lessons supplemented the labs while 49% disagreed. Students were asked what learning resources (other than textbooks) were used most by students during the year. Between 20% and 25% of the students (of 81 respondents) had never used the PLATO basic science lessons, journals/monographs or slide sets. Since it was known that students rely on textbooks as their primary learning medium, they were not asked how often they used this resource. In general, of all resources listed, journals/monographs and slide sets appear to have been used least frequently. It is interesting to note that 16% of all students responding to the questionnaire used the basic science lessons several times a week. This compares favorably to the utilization of other resources as only lectures were used several times a week as a learning resource by a greater percentage of individuals. Audiovisual media (e.g., films, videotapes) were employed by 38% of the students at least once a week.

Based on these data, it appears that, of all the learning resources other than the textbook, students utilized lectures. Other learning resources, including the PLATO basic science lessons, seem to be considered only as supplemental.

Project Accomplishments

When project accomplishments as they related to the specific contract objectives were reviewed, it was seen that no one objective was achieved completely. Many factors affected the project's functioning and impacted negatively on

the production of lessons. However, if project functioning is considered on a macro level with less emphasis on the specific contract objectives, a number of contributions can be identified.

One of the project's major accomplishments was that it demonstrated that PLATO IV was a usable and viable instructional system in the basic medical sciences. For example, student users at SBMS-UC felt that project-authored lessons were good supplements to the basic science curriculum. Students indicated that lessons helped them learn basic science concepts. They also requested that additional lessons be developed. Further, it was generally observed that some higher level of performance on examinations was achieved by lesson users as compared to that of nonusers. These differences, though for the most part, were not significant.

The project also demonstrated that many features of the PLATO IV computer system could be utilized effectively as a learning medium. Most of the courseware that was developed did, in some way, capitalize on the interactive, innovative capabilities of PLATO IV. For example, students indicated that they especially liked interactive lessons and lessons which graphically simulated complex biochemical reactions. Additionally, with the conversion of the Kansas pharmacology materials, the project exhibited that PLATO IV was an excellent host for lessons developed on another computer system. Based on the estimates of one project director, the Kansas materials were converted to PLATO at 10-20% the cost of creating lessons *de novo*. Further, there are some indications that the converted courseware ran more effectively on PLATO than it did on its parent system.

Another major contribution of the project was the role it played in the dissemination of information about PLATO IV to the CAI health science community. When the project began in 1973, little was known nationally about PLATO IV. At that time, the prevalent attitude was that the system was expensive as well as difficult to use and understand. However, partly through the efforts and demonstrations made by project personnel at national meetings and conferences, the PLATO IV system is now a sought-after commodity⁽²⁾.

Summary

In any educational development effort, it is sound practice to evaluate if particular approaches to the presentation of instruction are viewed favorably or unfavorably by the recipients of the instruction. In subsequent development

efforts, the findings of such an evaluation should be incorporated and should contribute to the improvement of the instruction. Relative to this study, it was found that students indicated preferences for lessons where emphasis was on problem solving. In terms of other educational practices, students responded that listings of objectives and key words, detailed feedback for incorrect responses, multiple paths through a lesson, a pre- and post-tests were important lesson components. Additionally, students noted that they liked being able to correspond on-line with authors of the lessons. This type of student-author "interaction" seems quite important as many opponents of educational computers charge that the use of computers dehumanize students because they remove them from exchanges with their instructor(s). A majority of students felt the computer lessons helped them to learn the basic sciences content and almost all felt the lessons were a useful learning tool. However, in almost all analyses of testing data, no significant differences in students' performances could be found.

It was also found that students viewed the style of the PLATO basic medical science lessons as compatible with the larger educational context within which they operate. It seems evident that, if an educational innovation is to be successful, it appears that close compatibility should exist between the philosophy upon which the innovation is based and the environment in which implementation occurs.

REFERENCES

1. Sorlie, W. E., J. D. Anderson, T. E. Gamble and D. K. Bloomfield, "A One Year Program in Basic Medical Sciences." *J. Med. Educ.*, 48:371-373, April, 1973.
2. Sorlie, W. E. and D. L. Essex, "The University of Illinois Basic Medical Sciences PLATO IV Project - An Evaluation." *J. Computer-Based Instruction*, January 1979.
3. Essex, D. L. and W. E. Sorlie, "Effectiveness of Instructional Computers in Teaching Basic Medical Sciences." *J. Med. Educ.* April 1979.

COMPUTER MANAGEMENT OF EDUCATIONAL COMPETENCY STATEMENTS

Fred G. Smith, D.V.M., Ph.D.
College of Veterinary Medicine
and

John S. Edwards, Ed.D.
Division of Educational Support
University of Georgia

Competency statements can be sorted and indexed by the computer so that instructors and students can use them more efficiently. This paper describes how one group of behavioral objectives has been coded and sorted by using a bibliographic indexing model. The cross-referenced lists of objectives have provided students with increased accessibility and given the instructors an effective way to evaluate which objectives are appropriate for each course.

Introduction

This report describes a technique for managing a corpus of five thousand competency statements pertaining to a veterinary curriculum. The division of Educational Support of the University System of Georgia Computer Network in Cooperation with the College of Veterinary Medicine, University of Georgia, developed a system for storing, indexing and retrieving data-element oriented statements such as bibliographic citations or, as described here, competency statements. The system concepts were similar to a bibliographic indexing model. Such an approach led to the development of a powerful tool for providing user access to competency statements.

The bibliographic indexing system provided sorted lists of competency statements that could be used by veterinary medical students, faculty, graduates, outside evaluators, and for certifying or licensing agencies. The system is ongoing in that different facets describing the courses and curriculums can be added to, deleted from, or changed just as the body of knowledge describing veterinary medicine is deleted from, added to, and changed.

Data Processing

The indexing system requires the storing a very large quantity of data. These data include the behavioral objectives and all the attendant references, resources, discipline and curriculum codes, etc. It also requires retrieving the data from a wide variety of viewpoints. The basic sciences faculty would need rather different access from that of the clinical specialists. Therefore, the data would need to be displayed in a variety of formats. In light of these requirements, it was felt that some sort of computer assistance would be beneficial.

In order to develop a management plan

for the study, two decisions were required immediately: 1) a computer file structure appropriate for handling the data associated with behavioral objectives.

The file structure which appears to lend itself most favorably to the needs of the data is that which is referred to as "data element oriented." The data element orientation of the file structure was chosen for two reasons: 1. To permit an open-ended file construction process. This technique allows unlimited addition to each file record as new data becomes available. 2. To allow use of an indexing model for storing and retrieving behavioral objectives.

The data element structure may be explained as a logical record which consists of unlimited discrete units (data elements). For the purposes of this study, the Magerian objectives model¹ was used to structure the objectives. This model consists of the statement of the task, the statement of the conditions under which the task is performed, and the criteria for evaluating the student's performance on the task. The choice of the Magerian model will not restrict the use of other models for defining objectives. The computer system data element oriented file structure can be modified by creation of new data elements at anytime. Thus, if a different objective model is preferred during subsequent development, the file can accommodate this by the creation of appropriate new data elements.

The flexible and open-ended file structure will allow maximum potential for accommodating additional information as suggested by subsequent investigation. Also, the data element structure permits a freedom in retrieving the information which will give great usefulness, in a wide variety of applications. Each data element is capable for functioning as an

index point (and retrieval point) for the objective. Thus, one can search the file for specific data such as type of objective, subject area objectives, etc. Using the report generator, indexes may be printed using any data within any data element as an indexing heading.

The second decision, definition of data pertinent to behavioral objectives, presented a greater challenge than had the choice of a computer system.

Logically, the problem involved two facets which had to be accommodated for successful retrieval of the set of objectives desired. Uppermost in consideration was the problem of identifying objectives by problem-solving task orientation, curriculum objectives course objectives, and by task orientation. In addition, the idea of sequencing objectives was considered of significance. Since behavioral objectives are created in a hierarchy, i. e., upper level objectives assume mastery of prior objectives, it was deemed essential that some means be provided to reflect this concept. In order for one to use intelligently the objectives, it will be necessary to follow a kind of path or track through the objective hierarchy.

After careful consideration of alternatives, it was agreed to follow a bibliographic indexing model for specifying the data pertinent to the objectives. The use of indexing techniques will provide solutions to both the problem of identifying objectives and their relationship to subject content and of tracking the objectives through their respective hierarchical structures. Implementing these techniques required that the data elements be specified and that the vocabulary which forms the content of each data element be standardized when possible.

The data elements identified to date generally fall into two classes: 1. Active type which are elements that contain knowledge to acquire, tasks to perform, evaluations, etc. 2. Passive type which contain descriptions of prerequisite skills and knowledge, resource indicators, cross-references to related objectives, etc.

Present Status of Project

At the present time, 4000 (80%) veterinary behavioral objectives have been coded and transferred to worksheets. About 3000 of these have been made machine readable.

After the first 1000 coded objectives had been indexed, it became obvious that several of the data elements could be effectively produced automatically by the indexing system. For example, the index produced by sorting nouns and noun phrases found in the objective, duplicated and even exceeded the resolution of certain preassigned classification categories. Thus, certain data elements could be dropped from the coding worksheet.

Summary

A bibliographic indexing system has been used to effectively sort the behavioral objectives of a veterinary medical curriculum. Such indexing has significantly increased the usefulness of the objectives and simplified the editing and updating required to keep the objectives current.

The indexing system allows an open ended classification system for behavioral objectives so that the entire system remains compatible with the needs of the veterinary curriculum.

References

1. Mager, R.F., 1962
2. McGuire, C. cited in "Levels of Learning" by Hilsenreth, 1975.

A COMPUTER MANAGEMENT SYSTEM FOR A MODULARIZED COURSE WITH HIERARCHICAL STRUCTURE AND OPTIONAL UNITS

ANN BREBNER, H. J. HALLWORTH AND W. D. WINN

The University of Calgary Faculty of Education

Many computer management systems have restrictions limiting them to use with specific course materials or requiring a linear path to be taken through the materials. This system, however, manages any course divided into modules consisting of a series of units arranged in a linear or branching hierarchy. Any number of units can be optional. The hierarchical structure and optional units allow students to choose different content and quite diverse paths through the learning materials, making the system particularly appropriate for courses for mature students. The system design is such that these additional facilities require little extra storage space or processor time. The design details and mode of operation are described. The system has been in use with a university level media course for two years. Its implementation is considered successful, and results relating to student achievement and attitude and to the analysis and improvement of the tests and learning materials are discussed.

MANAGEMENT SYSTEMS AND INDIVIDUALIZED INSTRUCTION

It is now widely recognized that there should be a move away from the traditional method of lecturing to students in large classes, towards individualized instruction where students work at their own pace and with materials best suited to their particular needs. However, those who attempt to implement courses in which the instruction is individualized soon become aware of several substantial problems.

A curriculum may be broken down into modules, and these in turn may be subdivided into units, each with a set of well defined instructional objectives. There remains, however, the problem of assembling sufficient good quality learning materials with which students can work individually, and of arranging these in such a manner that they form a well integrated course. Associated with this is the task of constructing adequate criterion referenced tests for each unit.

A further problem is to design and implement a management system capable of guiding students through the individualized curriculum. This may require placement testing of students, and always involves routing students through the learning materials, testing their understanding of such materials, recording the progress of each student, and providing reports as necessary. Such functions have rarely been performed successfully except by using a computer-based management system.

There is also the problem of evaluating the learning materials and their associated criterion referenced tests. Again, a computer is a useful

tool for the continuing evaluation and improvement of even a comparatively small system of individualized instruction.

A MODULARIZED MEDIA COURSE

The problem of developing instructional modules is illustrated by the course "Learning Technology in the Classroom". This is an introduction to instructional media and technology given in the fourth year of the Bachelor of Education program at the University of Calgary. It consists of three components: classroom lectures and demonstrations related to communication theory, media effectiveness and instructional design; laboratory activities involving the operation of equipment and production of instructional materials; and student projects, in which theory and practice are combined to produce both plans for mediated lessons and appropriate materials.

Although the course is optional, it nevertheless attracts a large number of students. Therefore, the laboratory portion cannot be scheduled in the timetable but operates continuously five days a week, allowing students to work their way through the various activities at their own pace at times convenient to themselves. Further, it is impossible to include in a single course the complete range of equipment and production skills that student-teachers might need to possess. As a result, the more important skills are identified as mandatory components of the laboratory with other skills being left as options. For example, every student has to learn how to use a single lens reflex camera, but only those who wish go on to work in the darkroom.

These logistical concerns, and the desire to provide some option to the students, led to a decision to make the laboratory portion of the course entirely self-instructional. Accordingly, funding was obtained from the Government of Alberta's Innovative Projects Fund in order to develop a hierarchical structure of modules each divided into smaller units, with the requisite teaching materials and criterion referenced tests. The intent was to lead the students through the laboratory activities with more economy of time and energy, whilst also providing sufficient flexibility for them to pursue in greater depth those activities that they found particularly interesting or useful. A final concern was to include a certain amount of cognitive content in the units in addition to motor skills, in order to facilitate the transfer of skills to unfamiliar equipment (Winn and Shieman, 1976). The laboratory component of the course now consists of a series of modules, each of which is made up of self-instructional units. These, together with associated multiple choice questions, were developed following recognized procedures of instructional development (Gagné and Briggs, 1974).

Within each module, some units are compulsory, some are pre-requisites to others, and some are optional. As a result, a student's route through a module must in some degree be controlled. This in turn implies that the instructor has to determine the hierarchical structure of each module at the beginning of the course.

Maintaining records of individual students, testing them, and directing them to the next unit, whether required or chosen, proved to be a difficult task for the laboratory assistant. It was therefore decided to use a computer management system for the laboratory activities, initially on a limited scale. The system was to contain three modules: dry mounting, production of transparencies for the overhead projector, and photography. It was required to perform the following functions:

- i. Provide a means for students to test themselves, on multiple-choice tests, as they completed each unit in a module.
- ii. Guide students, when they passed a test, to the next available set of units, thus keeping them to the sequence of units specified by the instructor, and letting them choose options at the appropriate places.
- iii. Keep a record of student performance (questions given, answers chosen, test scores, trials to criterion), and assign a grade if desired.
- iv. Enable course instructors to detect, through student success or failure, where the instructional units were not working well.

The problems of managing and evaluating an individualized course are best handled by a computer management system. Over the past decade a considerable number of computer management systems have been implemented and used quite extensively. The majority of these have had a somewhat similar structure. They have always contained a student file which records for each student identification data such as ID and name, and results of tests given on the learning materials. Associated programs store, edit and sort these data, route students through the materials, and print out lists both in detailed and summary format. In addition, some management systems have files containing the tests and, in a few cases, also the learning materials. These require further program segments to administer and score the tests and to output learning materials for the next unit.

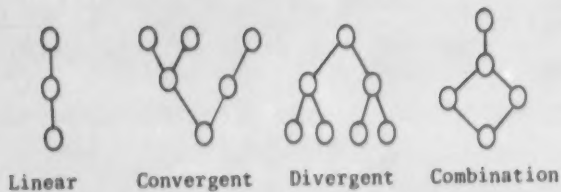
Many such systems are designed for use with a specific set of learning materials and have inherent features which prevent them from operating with a different set. Most require the students to follow a fixed linear path through the materials, thus individualizing only the time taken to complete the units.

For the course described above, such systems were obviously inadequate. A system was required which had, in addition to the usual features, provision to allow the students to choose, within certain limits set by the instructor, the order in which they wished to do the units, and also to decide whether or not to complete the optional units. A new computer-based management system was therefore designed and implemented, having all the required facilities. This system was developed for a time-shared machine, to be used on-line with a student taking a test whilst sitting at a hard-copy terminal.

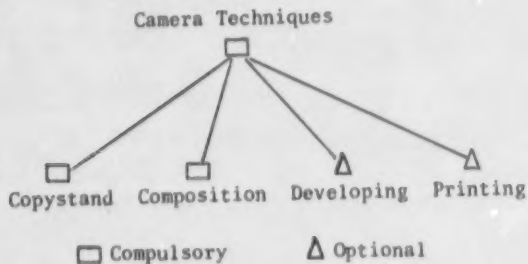
Although the system does not have the restriction that it may only be used in conjunction with a particular course content or structure, it is based on three underlying assumptions which must be met by any instructor wishing to make use of it.

The first assumption is that the course consists of a set of modules each containing units arranged in a hierarchical structure, as indicated above. The second is that for each of these units there is a test which must be satisfactorily completed before a student can proceed to another unit. Finally, the tests must consist of multiple choice items or questions to which the answer is a small integer number.

The actual hierarchical structures of the modules are defined by the instructor. They may be linear, convergent, divergent or a combination of these types, and may be entirely different for each module.



Any number of units may be optional. For example, the photography module of the course follows a simple divergent pattern and has two optional units:



Editing features exist which allow the instructor to add or delete units in a module at any time except when students are registered and active on the course. The system adjusts the configuration and makes some checks in an attempt to ensure that the structure remains valid. These checks, however, are not exhaustive and the final responsibility for the validity of the structure rests with the instructor.

Other parameters initially specified by the instructor are the number of questions to be given for each test and the mastery level required. Banks of test questions are then built for each unit. Again, editing features allow the instructor to add or delete questions or to make changes to existing items at any time.

At the beginning of each semester the IDs and names of the students are entered. A student may then start by taking a test on a fixed unit. If the test bank is relatively large each student's test will probably be different, since the required sequence of questions for each individual is drawn randomly from the bank without replacement. If the test is failed, the student is required to take it again when next logging into the system. The questions then received will usually be a combination of some given on the previous test and some new ones, as the test is comprised of items chosen in the same manner as before.

On successful completion of a test the student receives a list of all possible units which can be studied next. At any point in time, the units on such a list are governed by

the hierarchy and by the units which have already been completed. When the choice for the next unit has been made it cannot be changed and the student must pass the associated test before moving on.

Once all compulsory units of a module have been finished, those units from the next module which do not have pre-requisites are added to any remaining optional units on the list. In this way the management system, although controlling the overall direction of the course, allows students not only to take different paths through the learning materials but also to cover more material in those modules in which they have a particular interest or aptitude. Thus individualization is maximized.

Taking again the photography module as an example, the unit on camera techniques must be studied first; next there is a choice from units on the copystand, composition, developing, or printing. When both the copystand and composition units are completed the choice widens to include units from the next module. There are, therefore, 38 different ways in which the photography module may be approached.

DESIGN OF THE PROGRAM

The seemingly complex additional facilities available in this management system are achieved at little extra cost in either storage space or processor time. Although the system does contain one more file structure than normally found in a management system, this is comparatively small. The extra file, namely the hierarchy file, is the core of the whole system. By making use of the simple mathematical fact that the product of any given set of prime numbers is unique, the hierarchical description of a module can be packed into three locations for each of its units. Hence only 10 pieces of information are needed for each unit of the course. These are the unit name, its module number, the prime number of the module, the unit number, the prime number of the unit, the number of questions in the test bank, the number of questions to be given in a test, the mastery level as a percentage, an indicator showing whether the unit is optional, and the pre-requisite units stored as a product of their designated prime numbers.

The photography module is the third one in the course and is stored in the hierarchy file as follows:

Camera techniques	3	5	1	2	27	10	80	1	1
Copystand	3	5	2	3	32	10	80	1	2
Composition	3	5	3	5	28	10	70	1	2
Developing	3	5	4	7	22	10	70	2	2
Printing	3	5	5	11	15	10	70	2	2

The information in the hierarchy file, combined with that in a location for each module in the student record which is also used for the purpose of storing a product of prime numbers, is

sufficient to determine at any time those units a student has completed and also those from which the next unit may be chosen.

For example, if the product of primes in the student record for the photography module was 2, then this student would have completed camera techniques and the other four units would all be possible choices for the next unit. If, however, the product of primes was 462 ($2 \times 3 \times 7 \times 11$), then the student would have completed all the units except composition and would have no choice for the next unit.

Also contained in a student's file are the ID, name, current unit and a detailed record of each test taken including the number correct, the questions presented and the answers given. This latter information is being collected at the present time in order to assist the instructors in improving both the test questions and learning materials.

The questions are contained in a file structure. This consists of one file for each unit, allowing easy entry and deletion of units in the hierarchy.

PROGRAM OUTPUT

In addition to the usual type of summary and detailed lists on the performance of a group or an individual student, the instructor may at any time request information on the bank of test questions for a particular unit. This provides such data as the number of students passing the test at their first attempt, the number of students repeating the test, total number of repeated administrations, the average mark obtained and an analysis of answers given for each question in the bank. The summary table shows for each question the number of times it has been presented, the number of times each choice has been selected, and the correct answer. This information allows the instructor to identify those questions which are too difficult or too easy and often to discover deficiencies in the instructional materials.

For example, an early set of data from the photography module gave the following:

Composition

Total number of questions on file	28
Number of questions for test	10
Percentage correct for pass	70
Passed at first attempt	20
Number of students repeating	8
Total number of repeats	23
Average mark	6.51

Q	TP	1	2	3	4	5	CA
1	17	0	15	0	2	0	2
2	16	8	3	0	5	0	4 *
3	22	2	13	5	2	0	2
⋮							

25	17	1	0	14	2	0	3
26	15	5	0	0	10	0	1 *
27	15	14	0	0	1	0	1
28	21	11	4	1	3	0	1

From such data the problems associated with questions 2 and 26 can be investigated.

CONCLUSIONS

Continued use of the management system with the same course makes it possible to effect repeated improvements in both questions and teaching materials. The system was first tried experimentally during the summer of 1977. At this time, the computer analysis allowed some of the questions to be improved and some revisions were made to the instructional units. Starting in the fall of 1977 the system was used with one section of the course (approximately forty students), after which further refinements were made to the tests, and two more units were added to the photography module. In the fall of 1978, the system was used with all four sections of the media course, consisting of just under one hundred students.

All students succeeded in completing the necessary units with their accompanying tests. However, not all passed the first time. The photography module is again a case in point.

Unit	Camera	Copy-stand	Composition	Develop	Print
Passed first time	45.5%	54.3%	71.4%	48.4%	39.5%
Students repeating	54.5%	45.7%	28.6%	51.6%	60.5%
Average mark	71.5%	72.2%	65.1%	66.3%	63.3%

With the exception of composition, approximately half of the students failed the test on the first occasion. (The pass mark for camera techniques and copystand was 80%. For the other three units it was 70%). Inspection of scores for individual students showed that very few had to repeat the test more than three times, with most passing on the first or second attempt. A student failing a test was directed to see the instructor in order to determine the area of difficulty before repeating the test.

The data also indicate the difficulty of the various units. Further, the distributions of responses to individual questions indicate those parts of the instructional units which are not clear. For example, on question 26 from the test on composition, shown above, the majority of students answered that composition and harmony affect the dominance of a subject. This is incorrect and shows that the students had not learned the difference between balance, composition and harmony. A revision to the instructional unit has been made.

80 Written comments collected from students showed that the majority were enthusiastic about the system for two main reasons. The first was that using the terminal and the management system was an intrinsically interesting experience. Few of the students in the course had used a terminal previously and found the experience fascinating and informative, especially, as several pointed out, in a course on educational technology. The second reason was that the system allowed students to assess their own performance and keep track of their own progress.

For the instructor the results of using the system so far have generally been encouraging. Student data have permitted both revisions to the tests and to the instructional units. The problems that the students encountered with the system were readily correctable. A short instructional package on the use of the system needs to be produced, to minimize student difficulties in accessing the management system. Also, more feedback on student performance needs to be given at the end of each test. However, the system, and the modules it manages, are now sufficiently proven to warrant further developments.

The rest of the laboratory component is to be brought under the management system, and modules are to be developed for the more theoretical portions of the course. It is the intention that, eventually, most of this course will be modularized and individualized; and it is considered that, with a proven computer management system, this is now feasible.

Plans are also being made to use the management system in a modularized music course, in a course on surgical nursing in which computer assisted instruction programs will be teaching units, and possibly in courses in the Faculty of Humanities for written English and French. Experience to date suggests that this is the type of system needed for many courses offered to mature learners, and that it may without too many extensions be generalized for use in a variety of learning environments.

REFERENCES

- Gagné, R. M., and Briggs, L. J. Principles of Instructional Design. New York: Holt, Rinehart and Winston, 1974.
- Winn, W. D., and Shieman, E. Media and methodology considerations in the design of a modularized course. Proceedings of the Second Canadian Symposium on Instructional Technology. Ottawa: National Research Council Canada, 1976, 285-298.

COMPUTERS IN THE COLLEGE
CLASSROOM: AN EXAMPLE FROM
TRANSPORTATION GEOGRAPHY

C. Murray Austin
Department of Geography
University of Northern Iowa

This paper presents an approach to the study of transportation which is based on a series of interactive computer (H.P. 2000) programs. The use of the computer is different from most "programed instruction" with which the author is familiar.

The approach is to utilize the computer to explore both a data base and the meaning of several models used in the course. The approach has been found to stimulate imaginative analysis and to enhance intuition. One result of this approach has been a greater interest shown by the students in expanding their computer capabilities than had been the case under previous approaches taken by this instructor.

INTRODUCTION

Last fall (1978) I taught a course in Transportation Geography which, for the first time, was substantially focused around the use of an interactive computer. The course was intended to introduce intermediate and advanced undergraduates to the process, concepts, and procedures of transportation analysis and planning. Rather than provide a "cook book" approach the course was structured to have the students explore the issues involved in transportation analysis and planning and to experiment with the tools and procedures to better understand both their strengths and weaknesses. Many of the tools and procedures require lengthy computations and utilize a relatively large volume of data. Because such detailed and lengthy computation often result in arithmetic errors which tend to divert the attention of both the teacher and the student from focusing on the methodology and content and because, "in the real world" such techniques are rarely operated by hand I decided to build computer use into the course. As I developed the course under this precept (during the previous summer) I found myself modifying the role of the computer in significant ways.

The role of the computer as the course was eventually developed and taught became much more a central and integral part of the course. Based on the experience further changes will be made to increase the role of the computer in future offerings of Transportation Geography.

This paper will explain and discuss the

role of the computer as it evolved, evaluate the effectiveness of the approach and indicate the direction and nature of expected future modifications.

THE COMPUTER AND THE CLASSROOM

The computer system which I used was an Hewlett Packard 2000/ACCESS interactive computer with both C.R.T. and hard copy terminals. The programming language is BASIC. This type of system is appropriate for developing programs and approaches through which the student can interact with the concepts.

My past experience with the computer in the classroom was with what is commonly referred to as Computer Assisted Instruction (CAI). My perception of these approaches has been basically negative and I have concluded that CAI is, as I have observed it, often unproductive and even misleading. I base this on my observation that the highly structured and routine nature of the CAI programs often oversimplify issues, stifle creativity, and encourage guessing instead of reasoned judgments and conclusions. The computer thus becomes a device not understood by the student and the CAI programs become philosophical "black boxes." In preparing this paper I spent several days working with a variety of CAI programs available at our university and, except for programs reinforcing narrow skills, I have found that my earlier conclusions still hold.

The approach I developed differed substantially from the CAI programs I have seen. The computer becomes a time saving device

82 which not only allows, but assists the student in exploring tools, methods and concepts. My approach makes use of the computer's principal assets of high speed, accurate computation and storage of substantial volumes of data. It provides a more standardized structure for the student's exploration and thus given, the instructor some ability to focus that exploration while not reducing the role of creativity and imagination.

The use of the computer and the assignments on it were integrated with the lectures and discussion in the class. Attempts were made to fully explain what the programs did and how. The programs were designed to provide results for interpretation and analysis rather than answers to questions specified in advance.

THE APPROACH

The series of computer based assignments I developed for the transportation course were based on a hypothetical state (THE STATE OF CONTENTMENT) with a specified transportation network serving specified cities with given populations. The assignments were sequenced in order to focus the student's exploration in a desired manner.

Assignment 1: The first assignment was an attempt to have the students apply the concepts to the hypothetical state (some information of each city was provided) and using their intuition to explore the state and develop a set of reasoned expectations about the nature of transportation flows and needs for the state. This involved making use of information on the "type of city" (e.g. the state capital, a farm market center, a university town, etc.) and considering the types of factors which tend to influence transportation demand.

Assignment 2: The second assignment was designed to have the student explore several models of interaction which are used in analyzing and predicting transportation demand. These models which had been introduced in class were placed in programs and the student "played" with them. They varied the parameters of the data. The output they received showed the numbers which resulted when those models with the parameters they specified were applied to the data. This output was provided in both tabular and graphic form. The students were then assigned the task of interpreting, comparing, and contrasting the different models and asked to generalize about the relation of parametric variation to the values produced by each model.

Assignment 3: The third assignment involved the use of further hypothetical data on actual flows which they had to use to estimate the actual parameters for each model (using a regression program). They then had to choose the model they felt most appropriate and explain why. That model was then applied to estimate actual aggregate flows. The students were then to explore the implication of different temporal distributions of the aggregate flows. They had to discuss the implications of different temporal distributions on the requirements for transportation planning.

Assignment 4: The next assignment was to utilize the results and conclusions thus far generated to estimate transportation demand and to compare with more data on the existing transportation to identify potential and actual bottlenecks in the system. This was used to generate a general discussion of the effects of the demand pattern's variation from the capacity structure of the system. The final step of this assignment involved the students specifying growth rates and generating projections of future transportation demand.

Assignment 5: The fifth assignment using the computer had the students propose future changes in the existing transportation system to meet the projected demand. This was done in stages with each stage adding one or more complications to the system. The first stage was under no constraints and it resulted in the system fully serving expected demand. The second stage provided variable cost data and a budget and asked the students to experiment with system changes given the budget constraint. The changes finally proposed were the results of iterative changes in the absence of a decision making algorithm. This assignment formed the bases of several class discussions of the different needs of such a decision making algorithm and critical discussion of using the material thus far developed as the bases of planning.

Assignment 6: The sixth and final computer based assignment involved two steps. Social costs and variation in the social benefits of different components of the aggregate demand were introduced. Following further explorations of the implications of this on the "effectiveness" of various changes in the network the students were asked to provide a decision making algorithm (set of specific objectives and trade off criteria). This algorithm was then used to determine the changes in

the transportation network which would be proposed.

Each of the six assignments above were followed by thorough and lengthy class discussions. Each exercise and the following discussion lead the students to the next assignment. Thus the discussions of the deficiencies of the approach used in the fifth assignment prepared the students for the approach taken in the sixth assignment.

The series of computer based exercises described above clearly differ from the approach generally taken in CAI. Instead of promoting uniformity in answers it allows for interpretative and actual differences. Instead of existing as a separable component of the course or as a series of self contained canned exercises this approach depends on the classroom interaction. The role of the computer structure is to allow the student to experiment or "play" with the ideas rather than to stress a pre specified, one-route solution to problems presented.

EVALUATION OF THE APPROACH

Although this course has been taught only once using this approach and thus the basis for a thorough evaluation is somewhat limited, several observations and judgments can be presented.*

1. The students were more aware and appreciative of the "art" involved in transportation than they had been under more traditional approaches.
2. The students became more familiar with the computer, its necessity and value in transportation planning and demonstrated that interest by continued involvement with the computer following the completion of the course.
3. The students expressed both satisfaction and enthusiasm for the approach. The evaluation at the end of the course was unanimous in the expressed belief that more learning occurred later in other comparable courses.
4. My judgment, based on the written papers which concluded each assignment and the quality of class discussion is that the depth of conceptual learning was greatly enhanced by the computer structured explorations.
5. My judgment, supported by student comments is that the exploration of

* The statements concerning student responses and reactions come largely from their comments on the Student Assessment questionnaire used by the Department of Geography.

the concepts and models made possible by the computer led to a substantially higher level of understanding than otherwise was likely.

83

In general the approach appears to have been highly successful and was well received by the students. It had significant advantages over using a non-interactive computer approach as well as over an approach not focused on the computer. The simplicity of BASIC and the ease of learning to use the interactive system made it possible to use this approach with students whose previous experience with computers was limited at best.

I am convinced that the approach taken and the benefits gained would not have been feasible without using the computer.

FUTURE MODIFICATION

Through the operation of the course many minor problems were discovered. These problems are being dealt with. There were also several discoveries of opportunities which the computer based approach presented which were not exploited. The first of these opportunities was the apparent ease with which the student could be taught, without reducing the systematic content of the course, sufficient programming to provide even greater individual focus. To exploit this possibility, I am currently building a program which will allow students to input not only parameters, but also functional forms and then examine the common data base using their individual functions. A similar change is being made in the forecasting programs to provide the students greater flexibility.

A second opportunity exists in providing a broader range of programs, the purpose of each being specified to the student. This increase in the number of programs (in the analytic and forecasting stages of the course) will give the students a greater range for exploration and provide a greater basis for comparison of alternative approaches. The different programs will require different data sets and will vary in the degree of detail of the inputted data. Estimates of the cost of developing and maintaining the different data bases will provide the student a realistic view of the decision environment involved in transportation analysis and planning.

The final opportunity arises, not from within the course itself, but from the sequence of intermediate and advanced analytic courses in applied geography (economic, urban, transportation, and regional analysis courses). The fact that all of the students can be expected to take several other course in this set provides an opportunity to utilize the

84 computer to functionally connect the courses to a greater degree. Thus I am currently working to develop a sequence of programs for each course involved which are interrelated to the programs aimed principally at the student in the different courses. The result will be, hopefully, an heightened understanding of the interrelationships between these various subfields of geography.

CONCLUSION

The use of the interactive computer as a core tool for the Transportation Geography course has proved to be a major advantage both to the teacher and to the students. It has allowed the student to explore and manipulate larger sets of data than would be reasonable without it. It has provided more immediate results and greater opportunity to try various "experiments" with the data and compare the results. It has allowed me to introduce models and methods of greater complexity than before without the corresponding loss of comprehension. In general the interactive computer, when used in the manner discussed in this paper provides the students with a more realistic comprehension of the complexities and ambiguities, of the costs and the benefits, of the value and problems involved in transportation analysis and planning in the real world. This contrasts with the more typical oversimplifications and rigidity found in more conventional approaches to CAI.

CONTRASTS IN ACHIEVEMENT OF COMPUTATIONAL SKILLS BY LOW AND HIGH ABILITY SIXTH GRADE STUDENTS IN 1965 SMSG AND 1975 MODERN ELEMENTARY MATHEMATICS PROGRAMS

Ann D. Hungerman
Professor of Education
The University of Michigan

ANOVA and Profile analysis were employed to compare the computational skills of high and low ability sixth graders during a decade of efforts to individualize instruction. Comparison of lowest ability subgroups demonstrated no differences with statistical significance in their computational skills. There were differences with statistical significance favoring the 1965 SMSG highest I.Q. subgroup in fraction and decimal addition, subtraction and total. The differences favored the 1975 highest I.Q. subgroup in decimal division. Attempts to meet individual differences appear not to have been effective.

INTRODUCTION

Phase III analysis of the computational skills of sixth grade students in 1965 SMSG and 1975 Modern elementary mathematics programs was conducted to investigate further the differences between the achievement of high and low ability students which were reported in Phase II.¹ While Phase I reported differences generally favoring the 1975 Modern group in whole number computation and the 1965 group in fraction and decimal computation, Phase II revealed the 1975 group's advantage to be due mainly to the high I.Q. (equal to or above 106) subgroup. The low I.Q. (below 106) subgroups demonstrated remarkably similar scores in 1965 and 1975.²

The increasing concern about individual differences verbalized by educators in the recent decade would lead one to believe that the goal and the result of instructional efforts would be higher achievement by learners at both extremes of the ability scale. Teachers have long been criticized for "teaching to the mean" or the average child, to the detriment of both the gifted learner and the learner with real problems. In this Phase III analysis, the students were further stratified by ability level, assigned to one of four I.Q. subgroups using 91.5, 101.5, and 111.5 as cutoff points. The performances of the lowest and the highest of these four subgroups were analyzed using Analysis of Variance with pairwise comparisons and Scheffe allowances, Profile Analysis and Descriptive Statistics.

Lowest I.Q. Subgroups (I.Q. < 92)

Group	N	Mean	Min	Max
1965 SMSG	76	83.64	65	91
1975 Modern	76	85.16	74	91

(No statistically significant difference)

Highest I.Q. Subgroups (I.Q. ≥ 112)

Group	N	Mean	Min	Max
1965 SMSG	101	119.34	112	134
1975 Modern	95	118.67	112	136

(No statistically significant difference)

What follows is a description of instructional efforts to recognize individual differences in mathematics instruction over the last decade in a southeastern Michigan school system, a report of the most recent analyses of student performance on 80 computational items of the California Arithmetic Test, and some conclusions drawn from the findings.

INSTRUCTIONAL EFFORTS: Textbooks and Individualizing

In 1965, the original year of the study, the SMSG series had been in use for three years in the schools tested. Addison-Wesley was adopted for the 1965-68 period, and in the intervening years the trend has been toward increasing use of multiple texts. In 1975, four series had been identified as "basic texts" and all schools were encouraged to use all four of them. The reality of multiple

86 text use was reported by six teachers who reported use of all four texts (Addison-Wesley, American Book Company, Harcourt Brace and Houghton-Mifflin), four teachers who reported use of three of the four texts, and the eight remaining teachers who reported use of two of the four texts. "Mathematics Prescription Resource Document," a continuous progress format for mathematics education, K-12, is based upon those four textbook series and was developed by a 1971 project team, revised in 1972 by a summer team, and again in 1974 by the six-member Individualized Mathematics Study Committee, with the director of mathematics education as facilitator. The stated philosophy is "to teach each child in such a way as to allow him to succeed at his own rate."

When presented with a variety of ways to meet individual differences, and asked to check any which they were actually using in their classrooms, 16 teachers checked "continuous progress" (for 1-5 years); 13 checked informal ways as well as continuous progress; 2 checked that they meet individual differences without individualizing; and 1 failed to check any of the above.

FINDINGS³

Lowest I.Q. subgroups

1. There were no statistically significant differences between the 1965 SMSG and the 1975 Modern lowest I.Q. subgroups on any variable of whole number, fraction or decimal computation, evident in the Anova or Profile analyses.
2. Descriptive statistics revealed some slight differences which were not statistically significant: means were lower in fractions and decimals; maximum scores were higher in whole numbers and fractions; the minimum score was lower in whole numbers; the mean for whole numbers and the range for decimals remains unchanged.

Highest I.Q. subgroups

3. There were no statistically significant differences between the highest I.Q. subgroups for any whole number variable.
4. Statistically significant differences favored the 1965 SMSG highest I.Q. subgroup over their 1975 Modern counterpart in:

fraction addition .0081
9900 level Scheffe allowance
fraction subtraction .0001
9900 level Scheffe allowance
fraction total .0082
9500 level Scheffe allowance
decimal addition .0000
9900 level Scheffe allowance
decimal subtraction .0000
9900 level Scheffe allowance
decimal total .0003
9900 level Scheffe allowance

Profile analysis supported the advantage of the 1965 SMSG highest I.Q. subgroup in decimals; in fractions it displayed statistically significantly different profiles, but not statistically significantly different group differences for the combination of all four fraction operations. All of these differences were small.

5. Statistically significant differences favored the 1975 Modern highest I.Q. subgroup over their 1965 SMSG counterparts in decimal division (.0032, 9500 level Scheffe allowance).
6. Descriptive statistics for the 1975 Modern highest I.Q. subgroup listed lower means for fraction and decimal totals and a higher (not statistically significant) mean for whole numbers. The minimum and maximum scores were lower for fractions and unchanged (already the complete range, 0-8) in decimals.

Lowest-Highest Comparison

7. Descriptive statistics revealed an overlap of score ranges by the lowest and highest I.Q. groups as follows:

		Low I.Q. Max	High I.Q. Min
COMP-TOT (80)	SMSG	56	25
	TRAD	58	29
	MOD	59	27
WN-TOT (33)	SMSG	31	15
	TRAD	32	17
	MOD	33	19
FR-TOT (30)	SMSG	19	6
	TRAD	19	4
	MOD	20	3
DEC-TOT	SMSG	5	0
	TRAD	4	0
	MOD	3	0

CONCLUSIONS

The data reported here clearly indicate that the goal of higher computational skill achievement by most and

least able learners has not been reached. The 1965 SMSG and 1975 Modern lowest I.Q. subgroups demonstrated no differences in any area of elementary mathematics computational skills. The 1975 Modern highest I.Q. subgroup had the advantage only in decimal division (an atypical finding which may be attributed to recency of instruction), while yielding an advantage in addition and subtraction of fractions and decimals to their 1965 SMSG counterparts. Thus the least able learners have made no progress and the most able learners are doing even less well than ten years ago.

DISCUSSION

Most educators readily admit the shortcomings of recent attempts to reform curriculum through changes in content and patterns of organizing for instruction. Furthermore, the data on overlapping ranges of scores by the highest and lowest ability students offer us dramatic reminders that ability alone, even the highest, is not a guarantee of achievement, and that I.Q. scores, even the lowest, are not infallible predictors of failure. Sequencing the content is not enough. "Allowing" a child "to succeed at his own rate," is not enough. Future success in improving the computational skills of all-ability students is much more likely to be a product of more effective instruction and of more efficient review and drill procedures. This focuses the spotlight back upon the teacher and highlights the need for more effective staff development programs.

FOOTNOTES

1. Part I of this research report was published in the Mathematics Education Information Report, "Research Sections National Council of Teachers of Mathematics 55th Annual Meeting, Cincinnati, Ohio, April 20-23, 1977," ERIC-SMEAC: Ohio State University, Columbus, Ohio, December 1976.
2. Phase II analysis is reported in ERIC document ED 144839.
3. All data tables available; write to Prof. Ann D. Hungerman, 1302 SEB, The University of Michigan, Ann Arbor, Michigan 48109.

CRSIE - A COMPUTER RESOURCE SYSTEM FOR INSTRUCTIONAL EVALUATION

Randy Rousse
Ingham Intermediate School District
Office of Evaluation and Planning
Lansing, Michigan

CRSIE, is an instructional resource delivery system which utilizes the cost-effective development and applications of computer technology in the field of instructional evaluation.

Computerized Resource System for Instructional Evaluation - CRSIE, incorporates a new utilization of the dynamic-interactive applications of the well known CMI and CAI computerized instructional models. It is a system of, measurement-based, criterion referenced, objectives, forming a progressively comprehensive plan of computer simulation and evaluation techniques. Aims of these techniques are toward an interactive, individualizing, integrating and evaluative learning experience for its intended users. The primary focus would be toward the learner-user integrating and measuring his/her development of the necessary skills - cognitive, life, career, which describes any of several occupational and/or academic preparations for the "real world of work".

Objectives illustrative of the CRSIE system and concept are here outlined:

- . A resource data, storage-retrieval system which is program-training related and individualized.

- . Application of CRSIE and learner-user interaction will be through pre-programmed, descriptive computer simulations of the "real world of work" and/or academic preparations.

- . Learner-User interaction will model and allow development of life, career, and cognitive skills (dependent upon previously determined and established program goals and objectives) of self-discovery, role-playing and decision making skills, etc.

- . An Evaluative Core forms the heart of the CRSIE concept. Core design requirements are based upon previously determined and validated occupational/social factors which are descriptors of the "real world of work".

- . The overall role of the CRSIE concept is a targeted focus on helping the Learner-User to integrate skills and behaviors mentioned above, having pertinence to their chosen occupational/academic preference or preferences.

The paper and presentation is being made in hopes of addressing the growing stress on quality standards of education being delivered to students in today's schools. Decreasing enrollments, increasingly higher (and necessarily so) professional salaries, accountability demands, etc., are all contributing to a gradual erosion of traditional educational resources and dollars.

The CRSIE concept is seen here as bringing a new dimension by tying together the applications of cost-effective technology and evaluative planning/measurement of educational resources. CAI-CMI Systems are analogous to this type of computerized instruction.

Yet, CRSIE's distinguishing feature, 'adding' new dimension to previous computerized instruction systems, is its evaluative nature and emphasis. In short, this type of computerized evaluative instruction, or CEI system, is a new genre of computerized instruction. Thus, this greatly improves the potential to deliver a better level of quality in educational instruction to its students and other learner-users.

Users of CRSIE as an instructional resource delivery system can theoretically be from a number of varying fields, disciplines and institutional facilities dealing with the education

and skill development of its learner-users.

In this particular development outlined here for presentation, CRSIE is seen as being used on the vocational high school and/or technical career training center levels.

The CRSIE concept is actually an outgrowth of consulting work this presenter did for a vocational-educational career center in Ottawa County, Michigan in 1975 for development of an electricity-electronics vocational program. The evaluative, core concept, however, was not developed for implementation at that time. It is in fact the evaluative-measurement nature of CRSIE that establishes and gives value for its intended users as a cost-effective, educational resource delivery system of learning, instruction, and skill development.

In approach to application methodology, CRSIE is a criterion referenced, measurement based, individually scheduled performance, objective based, instructional system.

The data generated as a result of learner-user interaction with this CEI system is, by nature of the technology, more easily and quickly accessible and/or organizable. This data can then be used to form the "Evaluation Core". The core design requirements follow upon previously determined variables of measurement. These variables must be specifically selected and designed based upon program goals and objectives of each user of the CRSIE concept.

In this design and application of CRSIE, the Evaluation Core is intended to be composed of cognitive, occupational and social factor elements. These elements are not only descriptive of selected careers from the "World-of-Work" but, as well, form the program base-variables of measurement and evaluation, which comprise the Evaluation Core. Further, elements comprising the core can be integrated into a CRSIE system with these typical features:

- . integration with current on-going instruction programs (so long as they are objective referenced and measurement based)
- . instructor designed and implemented course elements and materials
- . accessibility and easy retrieval of elements pertaining to individual students
- . evaluative elements describing point-by-point learner-user outcomes

.Simulation Models built from descriptive program elements, etc.

.Learner-user element manipulation in simulation environment

.Learner-user, self-evaluation and measurement of course elements and criterion referenced objectives

In helping learner-users to integrate skill development in the cognitive, life and occupational areas, usage is made of the computer as a simulation and evaluative tool delivering increasingly more comprehensive skills instruction and development.

Necessary to this skill development is an environment of dynamic interaction between the learner-user and the computer as an instruction device.

The targeted focus, that is, the outcome of the CRSIE concept is directly realized at this point. By optimizing this interaction coupled with individual prescriptions scheduling, instructional drill and practice and evaluations, etc, the real potential of CEI instruction evolves. More specifically, a teaching-learning cycle² incorporating active student participation with an on-going evaluation component for teaching progress, has been adapted for instructional delivery by the CRSIE concept.

Some further outcomes and characteristics following upon the CRSIE concept are:

- Cost-effectiveness of technology with potential for mass instructional delivery and individual programming, recordkeeping, storage and retrieval, etc.
- Potential for creative management approaches of educational delivery
- Utility and practicality of individual instruction, training, reinforcement all saving upon classroom instructor time and energy. This gradually allows a new developing role of instructional manager of classroom and human resources. Thus allowing more time for actual prescriptive teaching and less time for coordinating the elements of instruction, i.e., materials, record keeping etc.
- New training and updating of teacher training to level of instructional managers³ of classroom resources
- Developmental, progressive and summative evaluation. Developmental testing and measurement; self-administered by learner user with teacher-administration access, storage and control.

¹"Propositions about Schools in 1980" Curriculum Structure-Mastery Education, Pp. 20, RBS Publications #OP-107; CHI-Present Trends and Future Directions, by Robert Scanlon.

²An Optimum Interaction Learning Laboratory by Raymond Wyman, N.East Regional ctr. for the Deaf. Educational Technology/August 1973

³Capital Area Career Center Concept, Ingham Intermediate School District, Mason, Michigan. i.e., Individualized, Modulized Instruction concept with classroom instructional managers.

- "Cross-integration" and management of learner-user evaluative results between teachers, administrators and students due to computer memory storage and access
- Learner-user integration and preparation of learning products as individual's progress develops
- Curriculum has potential of being auto-prescribing and advancing with immediacy of feedback as instructional process between student and computer takes place.
- "Organic" curriculum component. On-going development and programming of master course material, administratively self-evaluative of curriculum validity as it responds to learner-user growth. (in-step vs. out-of-step)
- New software packages allow simplified exposure, initiation and familiarization with programming of course materials for both instructor and students
- Greater job/field life-skills identification and integration. Growing importance due to specialization of fields and accountability of performance

CAI and CMI systems are analogous to this type of computerized instruction. Yet, as illustrated throughout the paper, CRSIE's distinguishing feature is its evaluative components and interactive nature with the learner-user.

CEI systems enlarge and expand upon the CMI and CAI concepts by focusing on this evaluative nature. Specifically, CEI's are evaluative processing of variables of instructional management and learning. The process is interactive with the computer as in CAI and allows progress measurement and record report as in CMI. Yet, CEI makes use of these computer systems and extends their effectiveness by adding diagnostic and thereby prescriptive dimensions to the instructional delivery process. The learner-user also has an extension over his/her own educational process. Their ability to manipulate this interactive learning environment, based upon their developing perceptions of the simulation model, allows a "quickenning" of the integration of course objectives and learner outcomes to be mastered. In summary, CEI systems should exhibit these main characteristics of differentiation:

- a) Diagnostic and prescriptive curriculum and scheduling
- b) Self-evaluative learner-user outcomes including curriculum development, tracking and feedback
- c) Simulation model development and interaction; allowing for "life-skills" interaction, decision-making skills development with increasing role self-awareness and knowledge

- d) Evaluative diagnosis available for instructional management of course objectives, on an individual basis, by the course instructor.

In short, this type of computerized evaluative instruction, as mentioned here, describes a new genre, a CEI system, of computerized instruction.

CVIS: WHERE CAREER EDUCATION
AND COMPUTER MEET

Dianne K. Garnett
Robert F. Garnett

University of Evansville
Evansville, Indiana

During the past ten years, interest has continued to increase in the use of the computer as a medium for the delivery of career guidance. The University of Evansville has been involved with computer-based guidance systems since the implementation of the Computerized Vocational Information System (CVIS) in 1973. Since that time research at U.E. as well as fourteen surrounding high schools and social agencies has shown CVIS to be an economical mode of delivering computer-based guidance which is enthusiastically accepted by students and counselors.

CVIS, a computer-based guidance system uses the computer as a tool to help individuals explore a wide range of occupations and educational opportunities. Computer-based guidance systems are considered a unique approach to assisting in the guidance and counseling field. Student/Counselor ratios are so high in most settings that counselors find themselves saddled with clerical and simple information dissemination tasks leaving them little time for other counseling/guidance functions for which they have been trained. Computers with their ability to store vast amounts of information, to retrieve that information quickly and with appropriate programming to simulate planned conversations, can provide counselors assistance in executing career education programs.

CVIS, is of particular interest because nationally it has the longest continuous use span of any computer-based guidance system. In January, 1967, CVIS began at Willowbrook High School in Villa Park, Illinois, with funding from the State of Illinois, Department of Vocational and Technical Education. Presently, school districts, colleges, and universities in all parts of the country have implemented the system and modified it to meet their local needs. CVIS not only provides a guidance function, but also includes the capability to add counselor-administrative functions and computer-assisted instruction.

The CVIS system is not designed to replace counselors. It should be viewed as a highly powerful technological tool which can assist individuals and counselors in the information collecting and sorting stage that should precede career

decision-making. The computer-based system, though easily operated by the individual alone, will not reach optimum effectiveness without complimentary career guidance services.

The University of Evansville began exploring computer-based guidance programs in 1971. After studying the various systems available CVIS was installed in 1973 since it appeared to best meet local needs. In February 4, 1974, CVIS terminals were placed in two local high schools as part of a pilot program. The University of Evansville's prime role relative to CVIS has been research, development, and evaluation for the Tri-State area.

A \$150,000 grant for the University's Project CVIS for 1974-1976 was obtained from the Lilly Endowment, Inc. These monies enabled the University of Evansville to expand and modify the data and scripts in CVIS to meet the needs of individuals in the State of Indiana and to place computer terminals which accessed CVIS in a number of schools and agencies.

To use the CVIS program, individuals interact with a computer by means of terminals called cathode ray tubes. These devices display messages on a TV-like screen to which responses are typed on a typewriter-like keyboard. The "conversation" is carried on by the transmission of data from the cathode ray tube over cables or phone lines to the central processing unit of the computer back to the cathode ray tube. . . instantaneously.

Individuals using CVIS may select any or

92 all of the nine subsystems presently available or under development.

Occupations: This program is designed to allow users to explore approximately 400 occupations. These occupations are catalogued by Dr. Anne Roe's classification system which includes six levels of training and eight categories of interest. Based on the choice made in each category, a list of occupational titles will be displayed. A description of the work tasks for any occupational title may then be displayed with the option of more information about work setting, promotional possibilities, training, personal requirements, salary or demand.

Four-Year Colleges: This branch of CVIS has three components. The first contains teaching scripts on general college information. The second, the college selection script, allows the user to enter any or all of seven college characteristics. Each time a characteristic is entered, the computer searches through 1600+ colleges and reports the number which satisfy the parameters. The third phase allows the user to receive five screens of information on any specific college: majors offered, admission requirements, and general information on cost, size, etc.

Trade and Technical Schools: This program parallels the four-year college section of CVIS. The first part of the subsystem, information about the selection of a technical or specialized school, has universal usability. The area dealing with selection of schools and detailed information on specific schools must be adapted locally. At the University of Evansville the more than 225 schools included in this subsystem are listed with the Indiana Private School Accrediting Commission or similar commissions in Kentucky and Illinois.

Apprenticeships: This subsystem of CVIS includes general information about apprenticeships with the ability to add localized information on apprenticeship opportunities in your area. At U.E. questionnaires were sent to companies and unions in southwestern Indiana to gather information on their apprenticeship programs. There is also an on-line quiz to help the user to learn about apprenticeships in general.

Financial Aids: This part of CVIS asks the user a series of questions about his interests and background. The result is a tailored list of scholarships, grants, and loans for which the student might be eligible. The entry of a given code number will produce a description of that financial aid.

Military Services: This subsystem provides information on military training provided by the various branches of the armed services. The military information script is divided into seven major areas:

1. draft information
2. opportunities for women
3. ways to view enlistment
4. qualifying for the military
5. types of jobs available
6. characteristics of military life
7. differences between the services

Local Job Information: This script provides tips for finding and applying for a job. Also, U.E. like most CVIS sites has localized a portion of this script so the user may access supply and demand data on various occupations as supplied quarterly by the Indiana Employment Security Division. State and national manpower projections are also available to help in understanding the future job market.

University of Evansville: This subsystem provides information on the degrees and majors offered at the University of Evansville and the specific courses required in each area. Any site installing CVIS can easily write scripts describing degrees and programs at their local colleges and universities.

Junior High: This subsystem is designed to allow junior high students to explore the world of occupations. The student is given an on-line interest inventory based on John Holland's work. This subsystem is supplemented by a series of audio-visual tapes describing the various jobs in each of Holland's groups.

The CVIS program was originally written to operate on an IBM 360 computer and was, therefore, compatible with the University of Evansville's IBM 370 model 135 computer. This made initial installation of CVIS relatively straightforward. CVIS was first installed to run in a dedicated partition as an assembler/BTAM program. Plans were soon made to reprogram the system since as first implemented CVIS required one-fifth of the University's computer resources making it expensive and leaving no room for additional on-line program development at the University. To eliminate these restrictions the University installed CICS which allows multiple on-line programs to operate in one partition. Therefore, work began in the fall of 1973 to reprogram CVIS to operate under the teleprocessing monitor CICS. Because arrangements had already been made with the two parochial high schools in Evansville to begin testing CVIS in February, 1974, the University

contacted other CVIS users nationally to find if any of them had begun a similar reprogramming of CICS.

St. Louis Public Schools had started on a similar project and were willing to share their work. This allowed the University to install the terminals in the schools as planned even though the St. Louis version was incomplete and did not follow the original design making conversion difficult and incomplete. The funding from Lilly Endowment enabled the University of Evansville CVIS staff to completely reprogram CVIS to operate under CICS keeping the modular design of the original program.

Upon receipt of the CVIS grant in July, 1974, IBM 3270 model 2 computer terminals were ordered. These terminals were scheduled to be installed in surrounding schools and agencies for a period of one year at no cost to the participating institutions.

Each school and agency was asked to appoint an individual to coordinate project at their respective institution. The Project CVIS staff then asked each coordinator to submit a written plan for integrating CVIS in their guidance program. After installation of the terminals, meetings were held periodically throughout the year with the coordinators from each site to discuss usage and any problem encountered.

Counselors at each site were encouraged to use and integrate CVIS as best suited to their institution. This provided a wide variety of plans with the Project CVIS staff serving in an advisory role to each site in contrast to training all counselors in one method of using CVIS.

During the second year of the project an evaluation plan was designed to determine:

A. Usage in both the school and agency setting with respect to:

1. Characteristics of CVIS users in each setting -- age, grade, and sex.
2. Frequency and length of time at terminal for typical student or adult.
3. Degree of orientation and follow-up by professional staff required for optimal impact.
4. Most frequently accessed sub-systems and by which group.
5. The most effective manner of incorporating the CVIS system into an ongoing program of guidance services.
6. Acceptability of CVIS as an on-line school administrative system.

B. The impact of CVIS in the school setting. Utilizing experimental and control groups a research design employing pre- and post-testing was to be employed to determine:

1. What effect utilization of the CVIS system had on those students who utilized it.
2. Differential rates of growth with respect to selected hypotheses for those students utilizing the CVIS system versus those students who do not use the system.

In the research design, experimental and control groups totaling approximately 140-150 students were included in the study. It was recognized that there would be some variation in approach utilized from one site to another and that the composition of the groups would differ. None the less there was to be a clearly identified experimental and control group at each site.

The authors experienced considerable difficulty very early during the operational year in exercising the necessary control over sites to assure at least minimal similarities among groups and in some cases the utilization of clearly designated "experimental" and "control" groups. Because of the freedom that had been given each site and because of the unique character and philosophy of each site, it became impossible to impose common restrictions and consequently the "experimental" groups were in fact in some cases career classes while in other cases traditional classes such as foreign language or mathematics and even in some cases they represented randomly selected students. Consequently, in the interpretation of the data, it was important to bear in mind that the word experimental was by definition of the site and for purposes of analysis they were grouped across sites regardless of the basis for inclusion at a specific site.

Pre- and post-testing was conducted by the CVIS staff utilizing the Career Maturity Inventory, research addition, (1973 authored by John O. Crites, published by CTE/McGraw). The basic hypothesis tested was that students who used CVIS exhibited a greater increase in career maturity as measured by the scales of the Career Maturity Inventory than those students who did not. Analysis of the results showed there was some evidence to support the hypothesis based on data which showed significance on four scales of the CMI when experimental versus control groups across all sites were compared. None the less, this must be viewed as a guarded conclusion because of the many sources of

uncontrolled error that could have influenced the outcome. On the other hand the overall data on utilization, student attitude toward utilization of an interactive guidance system, and counselor acceptability of computer-based guidance systems resulted in ample evidence to support that CVIS or similar computer-based guidance systems have a great potential in today's school setting. More specifically, it can be said that: (1) students are very favorable toward a computer-based guidance system as a source of easily accessible career data; (2) counselors are strongly supportive of having a computer-based guidance system in their building to support their overall career guidance efforts; (3) students who do utilize the system did demonstrate some increased growth as measured by the Career Maturity Inventory greater than those who did not; (4) the placement of the CRT is important and the most beneficial place appears to be a library or learning resource center rather than the guidance office; (5) there must be a clear designation of one person at each site who is administratively responsible for the implementation of the program and who will serve simultaneously as liaison with other schools as well as the central computer center site; (6) the guidance staff must have an in-depth understanding of the computer-based guidance system (it is dangerously easy to have a superficial understanding of what the program has to offer and, thereby, make erroneous decisions regarding its location, utility to various populations within the school or agency and most important of all integration into the on-going program of guidance services.); and, (7) there must be an ample period of time for creative development of the planned integration of the CVIS system into the on-going program of guidance services and a detailed plan listing responsibilities of all involved.

REFERENCES

- Chick, Joyce M. Innovations in the Use of Career Information. Boston: Houghton Mifflin, 1970.
- Garnett, Robert F. and Jones, Dianne K. Project CVIS: Final Report. University of Evansville, 1976.
- Harris, Joann, ed. Tested Practices: Computer Assisted Guidance Systems. The National Vocational Guidance Association, 1972.
- Harris, Joann. "The Computer: Guidance Tool of the Future," Journal of Counseling Psychology. Volume 21, Number 4, 331-339.

Office of Education. Computer-Based Vocational Guidance Systems. U. S. Department of Health, Education and Welfare, 1969.

EDUCATIONAL ADMINISTRATION,
EVALUATION, RESEARCH AND
PACKAGE PROGRAMS: AN
INTEGRATED EDP APPLICATION

S. James Zafirau, Ph.D.
Educational Research and Development Center
The University of West Florida

A programmatically-based and computerized management information system was created utilizing hundreds of selected demographic and psychometric measures for each youngster in an alternative educational and therapeutic wilderness camp system--including seven camps, three states. The original purposes of this MIS were to generate: A) quarterly and annual programmatic analyses for evaluating camps individually and together in terms of quantifiable successes and failures, and B) more rigorous research findings. The SPSS was selected because of its appropriateness and availability. To accomodate camp counseling information needs for individual youngsters, an integrated WRITE CASES procedure with unique print-out features was created. Teleprocessing and floppy disk storage is utilized.

Purpose

As with many educational enterprises -- perhaps even more so -- this project's educational and therapeutic wilderness camping program for youngsters was confronted with a vast assortment of student-camper data. This data ranged from birthdate to anecdotal counselor comments on the student-camper. There was no conveniently consumable consolidation or display of this wide array of data available: (a) on either a camper by camper basis for teacher and counselor use; or (b) on a grouped (by camp location, sex, age, and so forth) basis for camp director or central administrative staff use. Thus, crucial program evaluation, redesign and policy decisions had to be made more impressionistically than would have been possible with a well developed, data-based, management information system. This need became sharpened when viewed in the context of an educational system with relatively small school-camp components scattered from Vermont to Florida, and with student-campers whose unique profiles and educational needs required that a wider than usual array of measures and assessments be taken on them. The purpose of this project, therefore, flowed from the articulated need for the creation of some type of management information system as it related to programmatic versus solely business management concerns (payroll, purchasing, and so forth), which were to be serviced separately, as before.

Intended Users

There are a number of youngsters in each state for whom life and education within

the context of the traditional home and school setting is altered, for a variety of social and legal (as contrasted to physical and medical) reasons. Such youngsters may be excessively truant from school, ungovernable at home, or have run-ins with the law to such a degree that they come under the purview of some type of social service agency and/or court jurisdiction. The result may be a decision to place the child in some type of training school or care facility. The Eckerd Wilderness Camping Experience System, with teaching and counseling taking place within small groups in a wilderness camp setting, exists as an alternative to a training school placement for carefully selected youngsters who might otherwise be placed in such a facility. A myriad of screening data, treatment and assessment data, as well as follow-up data is gathered on each camper; prior to, during and following the stay in camp. Camp counselors and teachers need a complete and usable case file of information on each student-camper. Camp directors, central camp system staff, and various related state agencies need assorted information on a grouped basis.

Method

In early 1978, Eckerd Wilderness Experience Camp System representatives met with staff specialists of the Educational Research and Development Center (ERDC) of The University of West Florida concerning assistance in organizing and managing their programmatic information. As a consequence of numerous subsequent meetings and

96 the review of typical information collected on each student camper, codable versus uncodable (purely anecdotal) information was identified, and even this list of codable information was about twice the extent of the 324 codable variables finally entered in the system for each youngster. A time consuming process of analyzing each codable variable in terms of its functional importance was undertaken. Questions such as the following were asked and repeated to a variety of camp personnel and consultants: What is its (each variable's) importance and utility to the facilitation of the rehabilitation and education of each youngster? What are the individual camp as well as central camp system staff information needs for internal administration purposes? What information is needed for external purposes -- such as state reports, accrediting reports, and so forth? This required that not only the pertinent variables be identified, but also that the types of statistical analyses and output displays needed be identified, too.

Once these needs were determined, related data processing hardware, software and peopleware resources at The University of West Florida were reviewed in terms of their potential for developing and implementing the needed system. From the onset, advanced statistical routines were specified as needed to do extensive program evaluation related analyses on groups of youngsters. Only after the initial system was implemented did a request for a youngster by youngster case listing occur. Thus, the software packages most seriously considered were high level package languages such as the Statistical Package for the Social Sciences (SPSS), Bio-Medical (BMD) and Statistical Analysis System (SAS). None of these packages was available on the smaller University of West Florida computer, and so access to them was accomplished through the teleprocessing hookup to the Northwest Regional Data Processing Center in Gainesville, Florida. The SPSS program was finally selected for several reasons: (1) It provides all of the required statistical routines in very usable format, (2) the documentation and users manual are excellent, (3) it is probably the most widely used language currently in use in the social sciences, enabling a wide variety of potential users ready access to the data, (4) labeling features for use with the data are very good for such a package, enabling the creation of highly readable output, and (5) the efficiency of the routines is sufficiently good so as to make the development of a unique program from COBOL, FORTRAN, or the like clearly not justifiable in terms of comparative total development and operational costs.

At present, the teleprocessing system limitations confine the MIS use to batch

processing. Card or floppy disk input are possible, and output is rendered on multicopy paper forms from a high speed line printer, which are deburst mechanically for the several recipients of the identical copies. Total CPU charges for a typical multiprocedure, 9,000 plus line report is less than three dollars when done on an overnight production basis. This does not, of course, include the many other data processing costs which would be substantially the same, however, for any other language based output of similar complexity. The unique contents of this MIS presently include approximately 1400 masterprogram cards and nearly 1000 individual cases of 13 records each. This should double to nearly 2000 cases in the next five years.

A routine set of procedures based on a central set of forty plus variables is periodically run at the present on each camp as well as the camp system as a whole. This run includes the following:

A. Frequencies

Tallies for each value category are given for each variable plus percent breakdowns and a histogram. Descriptive statistics include mean, median, mode, range, minimum and maximum values, variance and standard deviation, standard error, kurtosis and skewness.

B. Breakdown

Causes selected variables to be broken down, one by one, in terms of state, camp, race, sex and age. The mean variable value number of cases, sum, standard deviation and variance are printed for each specified subgroup in the population. For the combined camps run, a one-way analysis of variance is run to compare how each camp is doing in terms of the other camps.

C. T-Tests

Student's t values are computed for paired samples of entry and midpoint scores attained by campers on IQ, aptitude and achievement tests, and self-concept and personality assessment scores, to determine where the camp program is succeeding or not. A correlation coefficient is produced, along with several other statistics. Probability levels are given.

D. Crosstabs

Pairs of variables whose values naturally occur in discrete categories, or are collapsed to such for convenience have their corresponding values displayed in a two dimensional array of frequency cells. Chi-square and other statistics are printed.

E. Write Cases

To service the needs of teachers, counselors and other camp staff to have an updated and integrated display of essential camper information on a youngster by youngster basis, a unique procedure was written. Output is displayed in a very readable format.

Individual variable values, as well as variable labels, are rendered in full alphabetic form. This requires the use of two or more program variables in order to display in alphabetic word form what appears to be one variable value (such as the race value of 'caucasian', which is actually three adjacent variable values which are each cued on the same, single column numeric value in the raw data file). This is necessary because the maximum width of an alphanumeric value in SPSS is four columns. The printing of decimal points is equally problematic. Although somewhat cumbersome and inefficient, the results are highly readable and valued among the camp system staff. This has led to plans for further extending the use of this procedure.

Special purpose runs are also frequently done to meet unique evaluation, report and applied research needs.

Results and Conclusions

This educational camping system's information needs were conceptually determined; codable and important variables were identified, as were the types of required displays and analyses, and a suitable package program was adapted through the creation of an extensive supplemental masterprogram. Procedures for collecting data, including the creation of special forms, were carefully developed, field-tested, refined and the residue has been made operational. Routine periodic reports are generated and disbursed appropriately, as are special analyses. Feedback from the field continues to result in further management information system refinements.

Plans to store all but those changeable parts of the program on hard disk and/or tape at the Northeast Regional Data Processing Center have been made to reduce transmission time. This improvement will occur as user requested changes in the basic system format subside. As is common, in this type of application, the client-user group has limited understanding of the technical requirements relating to different types of requests. In the longer range, it is hoped that a real-time, interactive, conversational environment can be realized so that feedback can be virtually instantaneous, creating the

framework for more extensive day by day decision making applications for this management information system.

REFERENCES

- Barr, A. J., Goodnight, J. H., Sall, J. P., and Helwig, J. T. A user's guide to SAS. Raleigh, North Carolina; SAS Institute, 1976.
- Health Sciences Computing Facility. BMDP biomedical computer programs: P-series. Berkeley: University of California Press, 1977.
- Nie, N. H., Hull, C. H., Jenkins, J. G., Steinbrenner, K., and Bent, D. H. Statistical package for the social sciences (2nd ed.). New York: McGraw-Hill, 1975.
- Zafirau, S. J. EWES management information system. Pensacola, Florida: University of West Florida, Educational Research and Development Center, 1978.

EFFECTS OF COMPUTER-BASED INSTRUCTION AND TUTORING IN AN ELEMENTARY SCHOOL SYSTEM

Lotus M. Knief

The University of Arizona

A school district implemented a tutoring program in reading, grades 1-3, and CBI in mathematics, grades 1-6, for low achieving students. Results indicated effectiveness for both programs and particularly indicated efficiency for the CBI program. Entry level behavior, sex, and self concept varied in influence by grade level and curriculum area.

When an area school district decided to implement computer-based instruction (CBI) in order to facilitate the basic skills learning of elementary school students who were in need of remediation, the district became aware that currently there is little argument that use of computers in education enhances the efficiency and effectiveness of learning. The reason seems to be the provision of improved learning conditions, especially in relation to the computer's ability to create a highly interactive learning environment where learning can be individualized (Bell, 1974; Educational Technology, Nov., 1974, entire issue).

In addition to the general effectiveness of computer use in instruction, research results also reveal specific learning outcomes. Studies consistently have shown performance increases when computer-based instruction (CBI) is used as a supplement to regular classroom instruction, particularly for students of low ability and for those requiring remedial learning. Vinsonhaler and Bass (1972) surveyed ten studies of the use of computerized drill and practice and concluded there is strong evidence of improved performance on standardized achievement tests. From an investigation of drill and practice in mathematics, Jamison, Suppes, and Wells (1974) developed an algorithm which predicted that a one percent increase in the number of computer sessions would result in a 5.72 percent increase in achievement!

Gains such as these are being achieved with continually decreasing computer costs. The current traditional educational delivery system appears to have matured to its natural limits and merely adding dollars has not resulted in improvement. Technology, however, promises to make a significant improvement in the current educational delivery system and should do so increas-

ingly as the theory of CBI is more fully developed.

Attitudes of both teachers and students must be considered and have been found to be favorable. Practically all of the research has included some type of attitude survey, and results have been positive. The general exception is an understandably negative attitude toward hardware failure. Among the few long-term studies, Porter (1959) found that elementary students retained their eagerness toward the use of machines even after a period of 34 weeks. Hess and Tenezakis (1971) found that junior high school students, after one or two years of computerized remedial mathematics learning, gave the computer very favorable ratings. An interesting finding by Feldman and Sears (1973) was that classroom behavior, as measured by a behavior survey (including satisfaction, motivation, social interaction, etc.), and achievement correlated less than is typically found in the subject area in which computer-based instruction was given.

Positive attitudes toward learning are extremely important for less able subjects. In fact, the computer turns out to be as fair and warm, in the perceptions of learners, as the traditional classroom teacher. A group of researchers (Hess et al., 1970) examined attitudes of low-income minority students who studied remedial mathematics by computer for up to two years. The exciting outcomes of the study were that the students perceived the computer to be almost exactly as warm as a human instructor. Even more critical, they believed the computer to be more fair, good, likeable, clear, and easy. Generally, Hess and his colleagues found computers judged as expert, fair, and emotionally neutral, and students trusted the computer--traits to be especially prized in

the instruction of those with failure backgrounds. Hess and his co-workers concluded that student-computer interaction has beneficial affective and social overtones in addition to the function of instruction.

From various types of investigations and also from studies of costs, the results are the same--quite positive instructional and cost results for the use of computer-based instruction. Furthermore, these results seem to hold for a variety of groups including disadvantaged urban, members of minority groups, rural populations, and also the deaf and multiple handicapped (Macken & Suppes, 1976).

As a result of outcomes such as these, it was decided that the school district involved in this study would implement a curriculum program which included an innovative CBI approach and expend some of its funds in a project where experience has shown that outcomes are likely to be successful. In the fall of the school year, the school district implemented a tutoring program and a CBI program in elementary reading and mathematics for those students in need of assistance.

Method

Subject

The participants in this study were students in grades one through six who had been identified by a combination of a needs assessment, testing, and teacher judgment as requiring assistance in the improvement of their reading and mathematics basic skills. Numbers of students varied according to the program(s) in which pupils participated.

Procedures

Needs assessment. A needs assessment was made in the spring of the year preceding the study. Relying heavily upon teacher recommendations, results indicated there were a number of students who could profit from basic skills learning assistance. The Iowa Tests of Basic Skills are given each fall in the school system, so those scores plus the previous needs assessment teacher recommendations were used in a cross-referral system of principal-teacher-school psychologist consultations to recommend pupils for remediation and hence participation in the project.

Parent involvement. The district Parent Advisory Council was informed of the screening and selection procedures. Additionally parents were involved in training conferences and advisory meetings and constantly were kept informed of the progress of the programs. Parental visits and special information meetings also were held.

Instructional system. The instruction in this project was structured as follows:

Reading: Tutoring in grades 1-3
CBI in grades 4-6
Mathematics: CBI in grades 1-6

In order to be able to implement computer-based instruction, it was necessary to obtain hardware, software, and courseware along with maintenance and technical support. After reviewing alternatives the Computer Curriculum Corporation of Palo Alto, California, was chosen for the following reasons:

1. Their package is being widely used successfully;
2. Evidence suggests that their products benefit both teachers and pupils;
3. Reasonable costs;
4. Adequate provision of implementation, training, maintenance, and support functions.

The computer curriculum in reading does not extend below grade level 2.5. Additionally, it was felt by district personnel that auditory feedback is essential in beginning reading. As a result, a tutoring program was instituted for assisting those in the early grades in reading.

The computer curriculum in mathematics does extend to first grade level, with practice in number concepts and beginning addition/subtraction combinations. Thus computer-based instruction was available for mathematics throughout all six grades.

The computer based instruction was an adjunct to regular classroom learning and did not replace teachers. Both the tutoring and computer-based instruction programs required some additional personnel, and these individuals were trained and supervised.

Orientations were held, and training sessions were conducted before school began for both the tutors and for those involved with the CBI. In both cases simulations of specific procedures were practiced, and the mathematics consultant and reading specialist began their consultations with and assistance to the teachers. These latter two persons acted in the same assistance role to teachers throughout the year.

Content materials. For the tutors it was necessary to set priorities and have structured lessons. The tutors reinforced:

1. Alphabet recognition (first grade only);
2. Consonant sounds (first grade only);

3. Dolch basic sight vocabulary;
4. Sight vocabulary from reading texts used in the classroom;
5. Phrase cards;
6. Oral reading from classroom tests.

Children were reinforced with points earned, stars, and occasional treats.

The CBI system is structured to allow a student to progress at his/her own rate in different areas of a subject. This is accomplished by the use of the strands. The elementary mathematics curriculum has been subdivided into fourteen strands or concept areas. Exercises in each strand are grouped into equivalence classes. Each equivalence class is a set of exercises with the same number properties and structure. A child advances through each strand independent of his/her progress in the other strands. The student does not receive an equal number of exercises from all strands; the proportion of exercises from each strand is adjusted to match the proportion of exercises covering that concept in an average textbook.

The computer reading program consists of one strand, Basic Sentences, at the lowest (third grade) level. As the student progresses, the Basic Sentences strand is replaced by 5 strands in various areas.

Testing Instruments

Three standardized tests were used in the program. The Wide Range Achievement Test (WRAT) provided the end-of-year achievement scores. The Iowa Tests of Basic Skills (ITBS) scores were used as part of the referral system. The Self-Esteem Inventory (SEI) was included as a measure of self concept at the request of school district personnel.

Design

The interest of the school district was in determining the achievement of program participants. Computations were made by school district personnel, following guidelines established by the state department of education. Results thus were calculated by finding the difference between the pre- and post-test grade equivalents for each student in each subject area and dividing that difference by the number of months between testings. Percentages of students making one month's progress for each month in the program then were determined.

The interest of the author was in determining the relationships of variables and the characteristics of students who participated in different treatments in the content areas. Regression analyses

were used, in each case regressing pre-test score first followed by other relevant independent variables. The hierarchical model was used to ascertain unique contributions and to allow sequential ordering according to major research interests. Grade levels were not used as a factor, since data inspection and gains calculations had shown no interactions but only straight line progression of achievement increase over grade levels.

Results

Results in terms of gains as calculated by the school district revealed that in both programs combined 74 percent of those in reading remediation gained one month for each month in a program. In mathematics 69 percent made a month of gain for each month in a program. This percentage is low, the district believes, because two primary grade teachers repeatedly did not see to it that students got to their mathematics CBI lessons. This kind of situation can occur after the start of a project despite very favorable cooperation and enthusiasm on the part of all others involved--a good example of field study hazards! Furthermore, despite the careful and thorough instruction, time spent in learning was not recorded as accurately as was intended.

Even in view of the above errors of omission and commission, the student gains and the reactions of administrative personnel, teachers, and the students themselves have been so favorable toward the CBI that it now has been extended to the junior high school level in the district and includes the language arts area.

Results of regression analyses in the two curriculum areas and two grade levels may be summarized as in the following four paragraphs:

Effectiveness was evident in both programs in that 74 percent in reading and 69 percent in mathematics made one month of gain for one month of instruction. Efficiency is indicated in that the CBI portion of these results was achieved in only ten minute a day sessions, adjunct to the regular classroom instruction.

Entry level behavior was of much greater importance in the tutoring program in reading than it was in the CBI program in this area.

The self concept of these subjects was not significant in either reading program nor in CBI mathematics at grades 4-6. The score, however, made a significant contribution in CBI mathematics at grades 1-3.

In mathematics, a significant ordinal interaction between entry level behavior and sex was found. High entry level students of both sexes achieved the same final performance levels,

but low entry level males exceeded the final achievement of low entry level females.

These results confirmed the expected efficiency and effectiveness of the use of CBI, to the extent that the school district has extended the CBI to the junior high level. At the same time outcomes indicate the need for further investigation of relevant variables and their interactions with self perception, sex and aptitude.

References

- Bell, F. H. Why is computer-related learning so successful? Educational Technology, 1974, 14, 15-18.
- Educational Technology, 1974, 14, 11, entire issue.
- Feldman, D. & Sears, P. S. Effects of computer assisted instruction on children's behavior. Educational Technology Review Series, No. 9, The computer and education. Englewood Cliffs, N.J.: Educational Technology Publications, 1973, 147-150.
- Hess, R. D. & Tenezakis, M. D. Selected findings from the computer as a socializing agent: Some socioaffective outcomes of CAI. Stanford: Stanford Center for Research and Development in Teaching, 1971.
- Hess, R. D., Tenezakis, M., Smith, I.D., Brad, R. L., Spellman, J. B., Ingle, H. T., & Oppman, B. J. The computer as a socializing agent: Some socioaffective outcomes of CAI. Stanford, California: Stanford Center for Research and Development in Teaching, 1970, Technical Report No. 13.
- Jamison, D., Suppes, P., & Wells, S. The effectiveness of alternative instructional media: A survey. Review of Educational Research, 1974, 44, 1-67.
- Macken, E. & Suppes, P. Evaluation studies of CCC elementary-school curriculums 1971-1975. CCC Educational Studies, Summer, 1976.
- Porter, D. Some effects of year long teaching machine instruction. In E. Galanter (Ed.), Automatic teaching: The state of the art. New York: Wiley, 1969, 85-90.
- Vinsonhaler, J. F. & Bass, R. K. A summary of ten major studies on CAI drill and practice. Educational Technology, 1972, 12, 29-32.

ELEMENTARY COMPUTERS
L'ANSE CREUSE PUBLIC SCHOOLS
ATWOOD ELEMENTARY

James Hollingsworth
Principal

Through the use of micro-computers elementary student learning can be facilitated by reinforcing and enriching their learning experiences. Students are learning with enthusiasm and achievement results are evident to the teacher and parent. Our students will be aware of computers and have computer literacy.

BACKGROUND

Located near Mt. Clemens, L'Anse Creuse Atwood Elementary, one of nine elementary schools in the district, currently enrolls 352 students (K-5), from an average economic community. Atwood first began its experience with computers in 1973, with one teletype terminal connected via telephone to the Macomb County Intermediate School District. Two more units were added in 1974 making use of the Hewlett-Packard Mathematics Drill and Practice, Spelling, and Word Puzzle Grids useable for spelling and vocabulary growth.

Using the Iowa Test of Basic Skills as a pre-and-post test, a study was conducted with two control groups to determine the effects of the computer on student academic growth. The study indicated students using the computer averaged .7 tenths more on the total battery than those not on the computer. The teacher survey suggested that a marked improvement in student self-image and confidence were important by-products. Parents indicated a marked improvement in school attitude by students who previously disliked school.

The availability of micro-computers was a welcome asset to our program. By phasing out the teletype terminals and using the micro-computer, we were able to eliminate the noise of teletypes as well as reduce costs. Telephone lines and costly maintenance contracts were eliminated.

PRESENT PROGRAM

The trend toward increasing use of computers is not likely to be reversed, therefore, with the aid of the TRS-80 Level II 4K Radio Shack Micro-Computers, we are making use of the latest technology.

Micros are moderately priced and can be used anywhere there is a suitable electrical outlet.

We presently have 6 TRS-80's placed in individual study corrals located adjacent to the Media Center. The program is supervised by the library clerk and several volunteer mothers. The 7th unit is used, as a portable unit available for total classroom or small group use. The 12" video monitor was replaced with an existing 23" black and white via conversion kit. The micros are used daily by 120 students or 34% of our enrollment. Student time is signed for by the classroom teacher who also selects the program from our current software supply.



Mrs. Miller, Volunteer Mother, assisting students Pam Schleicher and William Carp.

Our software tapes are produced by the Macomb Intermediate School District, under the direction of Mr. Tom Hartsig, Director for Computer Based Instruction. Input is received from the teaching staff.

Presently, we have tapes in the following areas:

MATH:

1. Number Line (Can be regulated as to maximum number requested - a flashing light moves randomly and the student responds).
2. Counting (1 tape larger numbers; 1 tape smaller numbers with varied distribution - student answers how many).
3. Multiplication Levels - #1, #2, #4 (From simple to difficult problems - stresses place value).
4. Tin Can (Probability math in volume for enrichment).
5. Drill (In simple addition).
6. Add 5 (Carrying and column addition, gradually more difficult - stress place value and carrying tens, etc.)
7. Subtraction (Stress borrowing and place value).
8. Prime Factor (Computes for students, factors any number they select under 100,000).
9. Expansion (Put in any denominators and numerators and number of terms and expansion is given).
10. Least Common Multiplier (Greatest Common Factor).
11. Addition and Subtraction (Drill and Practice Radio Shack Tape).
12. Addition and Subtraction Evaluation Tests (For addition - subtraction drills - Radio Shack Tape).
13. Digits (Given a number student applies place value).

LANGUAGE:

1. Grammar - Sentences changed each week - Parts of speech (4-5 grade).
2. Spelling - Each teacher, each grade level has an individual spelling list each week. (The number of words and intervals can be adjusted).
3. Phonics - Given Words (A flashing light moves randomly and asks if it is a short or long vowel - students answer s for short or l for long).

GAMES:

Hangman
Fly
Rectangles

As new tapes are developed for our computer-aided learning, several teacher ideas and interaction will produce better quality programs.

The advantage of micro-computing is that programs can be developed when there is a need for certain drill and practice or enrichment activities. Game can also play an important part in the total program.

Through this learning technique we hope to achieve these goals:

1. To reinforce, enrich and facilitate learning
2. To develop computer literacy.
3. To acquaint elementary students with a computer key board.
4. To allow students the opportunity to become aware of computers and their expanding role today.
5. To expedite learning in curriculum areas through drill and practice with immediate feedback.

Through surveys Teachers, Parents and the Students all say micro-computers are helping their learning. Through continual evaluation we will adjust and improve our program.

Many credits need to be given to our Superintendent, Mr. F. V. Pankow, for his encouragement and support. Thanks need also be given Dr. Roy Cloud, Curriculum Director for assisting in our project and Mrs. Martha Bradley, Library Clerk, and Miss Mary Cantoni, Librarian, for their devotion to the program. Appreciation is also expressed to Mr. Christopher Komorek, Manager, Lakeside Radio Shack, for prompt equipment delivery and advice.

Further information or visitations can be arranged by contacting:

L'Inse Creuse Public Schools
Atwood Elementary
James Hollingsworth, Principal
45690 North Avenue
Mt. Clemens, Michigan 48045

Telephone 949-4330

FILE-MANAGEMENT AND FORMATING TECHNIQUES FOR AN APL TEST GENERATION SYSTEM*

Daniel R. Lawrence

The National Technical Institute for the Deaf
Rochester Institute of Technology
Rochester, New York

ABSTRACT: The efficiency with which computer-generated tests are built and displayed can be increased appreciably by using file-management and formatting techniques easily deployed in APL. Through an APL program called INDICATOR, text files with varied dimensional components -- a file feature unique to APL -- are accessed for random test-item selection and problem construction. After interpreting code matrices, pre-established specifically for each problem stem, INDICATOR then uses a series of subprograms to manipulate the text for display according to specifications prescribed in the matrix coding. The result is a formatted problem represented as a printable text matrix.

Introduction

Computer systems for generating tests historically have been most beneficial in situations demanding many different tests over a short period of time with limited human resources available for their production. At Rochester Institute of Technology (RIT) in Rochester, New York, the Technical Mathematics Department of The National Technical Institute for the Deaf (NTID) services between 400 and 500 students per quarter. The department primarily functions as a learning center referred to as the Mathematics Learning Center (MLC), offering courses ranging from remedial algebra through calculus on an individual basis. All of the eighteen courses taught within the department depend on materials developed by our own faculty to supplement standard textbooks. Given the circumstances of the MLC (i.e., a large number of students taking tests individually, conflicting claims on faculty time, and ready access to computer support), it was determined several years ago that a computer test generation system was a desirable alternative to the traditional means of producing tests. Having gone through several evolutionary phases, computer test generation in RIT/NTID's Technical Mathematics Department will soon be handled entirely by a faculty-developed APL-based system.

Background

The Math Department's first experience with a Test Generation System (TGS) was with one imported from SUNY at Brockport, supported by an 1130-based IBM 1500 computer. The Math Department faculty created test-item pools for Algebra II-A and II-B (our two most populated courses) and used the system to generate all tests for these courses. (A test-item pool generally consists of five to ten similar test items designed to measure a specific behavioral objective in a particular course module.) In the summer of 1971, RIT replaced its IBM 360 with a Xerox Sigma-6 system to upgrade time-sharing and terminal capabilities. Though able to run independently of the Sigma-6, the IBM 1500 and its services were shortly thereafter abandoned for several reasons, including a paucity of replacement parts and a declining number of staff familiar with this computer.

The loss of the IBM 1500 also meant the loss of test generation capabilities via the Brockport TGS package. It was decided, though, that the observed benefits of a TGS justified the time and effort needed to build a new system, directly supported by the Sigma-6 and capitalizing on the already-created test-item pools. The new "in-house", FORTRAN version was designed and implemented by one of RIT/NTID's computer personnel and was more

*This paper was produced in the course of an agreement with the Department of Health, Education and Welfare.

closely directed to the needs of the Technical Mathematics Department than the one brought in from Brockport. At the same time, turn-around time for printing the tests markedly improved and, even though, during the summer of 1976, RIT purchased its present computer (a Sigma-9 system) to replace the Sigma-6, no modifications in the TGS were required. Alas, however, the direct involvement of the Math Department in the development of this TGS was limited to simply supplying the content for the test-item pools.

As the number of students served by NTID's Mathematics Department continued to grow and more frequent requests for generated tests were made, the lack of control our department had over the FORTRAN TGS became increasingly apparent. Immediacy in getting tests printed when needed was severely lacking. When curricula changes resulted in test-item modifications, corresponding corrections in the TGS data bank, because of the manner in which the files were structured and stored, could not be made without difficulty. Finally, documentation notwithstanding, only the author of the FORTRAN TGS was really familiar with its inner workings, and hence, when the system broke down, his services and his alone were needed to correct the problem. Clearly, it was time for the Technical Mathematics Department to investigate the idea of building, maintaining and deploying its own Test Generation System -- one over which we would have greater control and could more easily access when it was needed.

The APL TGS

Development of the APL TGS commenced in early 1977, in coordination with a recent revision of curricula for the Tech Math sequence (college-level algebra, trigonometry and analytic geometry). It was decided that since tests for the new modules in these courses would have to be constructed anyway, our interests could be served best by building test-item pools into a computer data base and then developing the new APL Test Generation System around it.

The FORTRAN TGS builds keyed text files line by line from card or tape images, checks the files for proper sequencing and then accesses them for printing. In contrast, the APL TGS accesses pre-constructed APL files -- hence, no need for checking sequence -- and after making appropriate formatting modifications to the text, prints it. In the FORTRAN version, with the exception of spacing between problems, all formatting must be done on the cards or tape. A problem is printed as an Nx80 matrix, where N is exactly the number of lines in the keyed file allotted to that specific test item plus the blank lines appended by the program to provide the student working area to solve the problem. Since each program is self-contained in its own matrix, the files must necessarily repeat the problem stem, spacing within the problem and any other formatting features common to all test items in that pool as many times as there are items in the pool. (See Figure 1 below.) The APL TGS does not have such repetition in its file system.

Figure 1. Sample of FORTRAN TGS keyed file contents (test items 3L1-9 and 3L1-10 of Algebra II-A, Module 3). (Compare with Figures 3 & 4.)

```

1350.000      1610 1 S
1351.000 3L109 1610 200A      6 . -2
1360.000 3L109 1610 300A
1361.000 3L109 1610 400A
1362.000 3L109 1610 500A
1363.000 3L109 1610 600Q      SOLVE BY FACTORING **
1364.000 3L109 1610 700Q
1365.000 3L109 1610 800Q      2
1366.000 3L109 1610 900Q      X - 4X = 12      X=      . X=
1367.000      1620 1 S
1368.000 3L110 1620 200A      3 . -8
1369.000 3L110 1620 300A
1370.000 3L110 1620 400A
1371.000 3L110 1620 500A
1372.000 3L110 1620 600Q      SOLVE BY FACTORING **
1373.000 3L110 1620 700Q
1374.000 3L110 1620 800Q      2
1375.000 3L110 1620 900Q      D - 24 = -5D      D=      . D=
1376.000 3L110 16201000Q
1377.000 3L110 16201100Q
1378.000 3L110 16201200Q

```

VINDICATOR[]V

```

V INDICATOR PROB1;FILL;G;SIGNS;X
[1] SIGNS←(' '=SIGNS)/SIGNS←.PROB1[SUB[K];13;13];PROB←1 3+PROB1[SUB[K];:]
[2] SIGNS←(' '*X)/X←.MARKS[SUB[K];:];←(1+126)*1WH←'T'
[3] FILL←FREAD 2,SUB[K]+1;G←I←1
[4] L1←(126)+(SIGNS[G]=' IDBMPGRTAEHOSLF')/16
[5] →DO;NOTHING PROB
[6] →DO;INSERT PROB
[7] →DO;DELETLP PROB
[8] →DO;BITDELP PROB
[9] →DO;MOREFILL PROB
[10] →DO;PRINT PROB
[11] →DO;GRAPHNOTE PROB
[12] →DO;GRAPE PROB
[13] →DO;TABLENOTE PROB
[14] →DO;TABLE PROB
[15] →DO;EXTRALP PROB
[16] →DO;HAMOLF PROB
[17] →DO;ONELP PROB
[18] →DO;SHORTNL PROB
[19] →DO;LONGNL PROB
[20] FLAG PROB
[21] DO:→L1×1(ρSIGNS)≥G←G+1
V

```

Figure 2. APL program INDICATOR which call subprograms to make formatting modifications on the "raw" problem stem.

The special features of the APL TGS are embodied in an APL program called INDICATOR which in turn calls up to sixteen subprograms. (See Figure 2 above.) INDICATOR builds the problem as a text matrix to look much like that built by programs in the FORTRAN TGS -- but in a much different way. Depending on the course and module for which tests are to be generated, a particular APL text file is opened for access. One text file exists for each module of each course and has as its first file component a three-dimensional array composed of the basic stems of all the problems to appear on the printed test for that module. The remaining ordered components, one for each such stem, contain the pools of "fillers" used to complete the stem. (Components of an APL file consist of variable-length data together with information on the structure of the data.) Once the "raw" stem from the first file component has been retrieved and a random selection of the test-item "filler" from the corresponding pool file component has taken place, the stem and filler are given over to INDICA-

TOR. (See Figures 3 & 4 below and 5 & 6 at top of next page.)

The first three rows and columns of each Nx75 stem matrix are reserved for code letters or characters to be analyzed by INDICATOR. (See Figure 7 at bottom of next page.) Each of the sixteen possible characters which could appear in this 3x3 code matrix, including the blank $\emptyset$ or ' ', causes a particular formatting subprogram to be activated. For example, "I" is the code character which activates a subprogram called INSERT. INSERT takes the stem matrix, from which the first three columns have been dropped, and replaces in order all asterisks (*) in the stem with the filler randomly selected from the appropriate pool file component. The appearance of the code letter "P" in the code matrix would cause INDICATOR to call PRINT which would append the filler as additional rows to the stem matrix, as opposed to inserting it within the already-existing text. Other code characters cause notes for the student or pre-constructed numberlines to be added to the stem matrix; some simply function to instruct the addition (or deletion) of

Figure 3. Sample of contents from first component of APL file for Alg. II-A, Module 3 (stem for all test items in item pool 3L1).

PH SOLVE BY FACTORING --

Figure 4. Sample of two "filler" items to be appended to the problem stem in Figure 3.

2			
X	-	4X	= 12
2			
D	-	24	= -5D

Figure 5. Sample of contents from first component of APL file for Tech Math II, Module 2 (stem for all test items in item pool 2A1).

```

ICE      <OBJ. 2A1>
EF FOR THE FUNCTION  $Y = \sin(*****)$ , STATE EACH OF THE FOLLOWING:

AMPLITUDE = _____. FREQUENCY = _____. PERIOD = _____.
DISPLACEMENT = _____. ENDPOINT = _____.

NOW SKETCH 'ONE CYCLE' ON TRIG GRAPH PAPER.

```

blank lines or rows to the matrix to provide an appropriate amount of space for the student to display his/her work in solving the problem.

INDICATOR looks at the code matrix one character at a time, modifying the stem matrix in some way for every non-blank character which appears in the code matrix, in the order of appearance of the characters within the matrix. Consequently the stem matrix might not be changed at all or, exceptionally, as many as nine manipulations or formatting modifications could be made in what is to be the problem text. One code letter of special note is "F". In the event that a particular mathematical symbol is needed to complete a problem and give it full meaning (e.g., pi, infinity, square root, summation, integral, standard deviation, etc.), with no such symbols readily available on the computer terminal or line printer, "F" causes INDICATOR to call FLAG. FLAG then effects the printing of an asterisk (*) immediately to the right

Figure 6. Sample of the "filler" items to be inserted into the problem in Figure 5.

```

22X -
32X - 2
23X -
33X - 2
34X - 2

```

of the problem's identification number on the generated test. (When the computer-generated tests are being prepared for student use, the appropriate missing symbols are filled in with pen.) After the insertion of any necessary flagging marks,

```

ICE
EF

```

Figure 7. Matrix of code characters which cause INDICATOR to activate specific formatting subprograms.

Figure 8. Formated problem for Tech Math II, Module 2 (test item 2A1-4) as printed out on the line printer. Inclusion of the hand-written symbol π makes the problem ready for testing purposes. (Compare with Figures 5 & 6.)

```

1.      <OBJ. 2A1-4> *

FOR THE FUNCTION  $Y = 3\sin(3X - 2\pi)$ , STATE EACH OF THE FOLLOWING:
AMPLITUDE = _____. FREQUENCY = _____. PERIOD = _____.
DISPLACEMENT = _____. ENDPOINT = _____.

NOW SKETCH 'ONE CYCLE' ON TRIG GRAPH PAPER.

(IF YOU DO NOT ALREADY HAVE GRAPH PAPER, PLEASE ASK FOR SOME NOW.)

```


the problem is printed as an Nx72 text matrix. (See Figure 8 on preceding page.) The process is then repeated until all the "raw" stems from that particular module's first file component have been accessed, formatted and printed, at which time generation of the test is complete. Iterations continue until the total number of tests requested for that module have been printed.

The APL Test Generation System is designed to be easily accessible and flexible enough to handle requests for any number of tests. It can service immediate requests for one or two tests which can be printed out right at a local terminal, or for larger numbers of tests, production of which would be very inefficient at a terminal, such requests can be accommodated through the use of the line printer. In either case, the test generation is initiated through an interactive APL program which allows the user to select the course, module within that course and the number of tests to be generated in any production. The APL TGS is completely at the disposal of any department staff member needing to use it.

Future of the APL TGS

To be sure, the APL TGS is not the solution to all problems related to test formatting and production by computer. A glaring lack in the system is its present inability to print stored answers to the individual text items at the end of the test. (One of the more attractive features of the FORTRAN TGS is its ability to do just that.) The answer page could be detached from the test before the test is given to the student and then used later by the instructor in grading the student's test. Answer keys are available for tests produced by the APL TGS, but they are not computer-generated and are not nearly as easy to use as those produced by the FORTRAN TGS and printed specially for the tests to which they are attached. Though computer-generated answer keys are certainly not as crucial to the smooth operation of the evaluation process in the MLC as computer-generated tests, the added convenience they provide the faculty in test correcting makes them still very desirable. Modifying the APL TGS to enable it to print answer keys with the tests is now the major priority in efforts to upgrade the system's capabilities.

At present, permanent disk space -- or rather lack of it -- poses another problem. Disk space limitations do not permit any more than two or three courses to be stored on line at the same time. As the TGS data base continues to expand to include test-item pools for more courses, the procedures for getting appropriate files put on line will become increasingly burdensome. Quite simply,

this means making efficient use of these files while they are on line.

The APL TGS generates all tests for Tech Math II & III currently, and by Fall, 1979, will have taken over the generation of tests for Algebra II-A and II-B as well. The expectations within the Math Department are that sometime in the near future tests for all of the most frequently taught courses will be generated by the APL TGS. Right now, the primary obstacle to this happening seems to be the lack of faculty time to build the test-item pools and construct the accompanying files. Whether the APL Test Generation System is the Technical Mathematics Department's ultimate answer to test generation or just another link in a long evolutionary chain remains to be seen. For now and the foreseeable future, however, it has -- or soon will have -- the capacity to function in every way one could reasonably expect.

INSTRUCTIONAL DECISION SUPPORT SYSTEMS: DESIGN CONSIDERATIONS

William C. Bozeman
The University of Iowa

The advent of machine age science has spawned a host of interdisciplines such as systems analysis, operations research, and human-machine systems. Closely associated with these are certain technological and management science implications for education. A topic which has received insufficient attention is the consideration of design variables in systems of instructional decision making. This paper presents a conceptual framework for decision support systems which addresses human factors, organizational, and contextual dimensions in addition to the technical/engineering aspects of information systems design. In this context, technology is viewed as a process and not only a product.

Introduction

Educators concerned with the enhancement of instructional management functions have demonstrated an increased interest in technological systems during recent years. This interest has, to a great extent, been a function of the revolutionary advancements in EDP. In less than two decades the computer has experienced a metamorphosis--a transformation from a behemoth weighing tons to a device roughly the size of an attache case. More remarkable than its diminutive size has been the increase in computer power and memory capabilities which have been accompanied by a decrease in cost.

Computer-based educational systems may, of course, serve a host of functions related to instruction but two general classifications are appropriate for the purposes of this paper--computer assisted instruction (CAI) and computer managed instruction (CMI). Though the latter is the primary focus in this paper, it may be useful to distinguish the two systems. A succinct but prescriptive definition of CAI is offered by Thomas (1979):

Computer-assisted instruction (CAI) is a pedagogical technique which a student interacts with instructional stimuli at a computer terminal, usually on a one-to-one basis. The CAI rubric includes (1) drill, in which the student responds in a

rather quick fashion to brief items or questions under a sort of "flash card" format, (2) practice, in which the student answers more complex questions which may require some off-line computation or the completion of multiple steps in the problem solution, (3) tutorial programs, which resemble programmed instructional texts in that paragraph material, interspersed questions, and response sensitive branching are present, (4) simulations, which model phenomena of a complex nature and in which random events are typically introduced, and (5) problem solving programs, which eliminate complex calculations to foster student understanding of principles and rules.

CMI is a decision support system or instructional management information system (IMIS) designed to assist in the management functions (e.g., planning, guiding, organizing, scheduling) associated with instruction, particularly when individualized schooling is a goal. Such devices extend far beyond data storage retrieval. A complete CMI/IMIS will certainly include data management capability, but will also perform analysis of data, research conclusions, and offer prescriptive recommendations. In order

for such a system to reach full potential, it must be designed with the higher-level purposes of instruction in mind. Typical CMI process elements include student data base maintenance, identification of instructional needs, grouping of students for instruction, reporting and profiling, and curriculum evaluation.

CMI/IMIS Design Considerations

Much like the business and industry management information systems (MIS) which emerged in the early sixties, CMI systems have traversed phases characterized by both enthusiasm and disappointment. The present phase is characterized by cautious re-appraisal of the basic concepts. Many system failures can be traced to three sources: (1) absence of meaningful involvement of the instructional staff during the planning and design process; (2) lack of adequate teacher and administrative knowledge regarding system use, operation, capabilities, and resource requirements; and (3) emphasis on the hardware/software/engineering aspects and components of the system.

Such problematic areas and design weaknesses may well be a function of the absence of a conceptual framework for a CMI/IMIS. Mason and Mitroff (1973, p. 475) offer a prescriptive definition and framework for a MIS which incorporates five important dimensions.

An information system consists of, at least, a PERSON of a certain PSYCHOLOGICAL TYPE who faces a PROBLEM within some ORGANIZATIONAL CONTEXT for which he needs EVIDENCE to arrive at a solution when the evidence is made available through some MODE OF PRESENTATION.

The variables embraced by this definition can offer important guidelines and direction in the planning and design of a CMI/IMIS. These five variables are addressed one-by-one in the following sections.

Person

The person, or system user, may well represent the most important design consideration and is given primary attention in this paper. A recent study by the author (1978) examined the relationships between the psychological or personality types of a set of users of a CMI system and certain factors associated with the utilization of the system.

The factors addressed were (a) user apprehension associated with human-machine systems, (b) confidence in the system, and (c) perception of usefulness of the system.

The theoretical framework for the study was Carl Jung's theory of psychological types. Essentially Jung's theory posits that certain apparently random variations in human behavior are, in fact, orderly and consistent. These variations, according to Jung, are due to differences in the way people use their perception and judgment (Jung, 1923).

"Perception" refers, in this context, to our process of becoming aware of things, people, occurrences, or ideas. "Judgment" refers to our processes of coming to conclusions about what we have perceived. If people systematically differ in what they perceive or in the conclusions they reach, they will show corresponding differences in their reactions, values, needs, and motivations. These basic differences in perception and judgment concern the way people prefer to use their minds, thus affecting their behavior. Different decision making styles are certainly characterized by different individual behaviors.

According to Jung (1923), there are two contrasting methods of perception;

1. Sensation is the way we realize that a thing exists. Our reliance on it is based primarily on the data we receive from our five senses about the objectively-real world.

2. Intuition is influenced by ideas or associations that our unconscious tacks on to our perceptions of reality. Intuition is exhibited when we attempt to see through facts and extrapolate beyond them, by our ability to have hunches about things and by our placing more interest in a thing's possibilities than in its present reality.

Sensation and intuition are opposed. One who senses future possibilities through intuition is likely to lack an appreciation of concrete reality as perceived through the senses (Myers, 1962).

Jung (1923) also indicates the existence of two contrasting methods of judgment:

1. Thinking is the way we tell what a thing is. It is essentially a logical process aimed at deriving impersonal findings. It relies primarily on cognitive processes and exhibits a tendency toward our making abstract, true/false judgments and our developing formal systems of reasoning and systemization.

2. Feeling is our process of appreciation, bestowing on things a personal, subjective value. We rely primarily on our emotional processes, and our evaluations tend to run along the lines of good/bad, like/dislike, pleasant/unpleasant, or acceptable/unacceptable.

Either judgment method can occur with either method of perception, creating four possible combinations.

Sensing plus thinking	Thinking plus intuition
Sensing with feeling	Feeling with intuition

The instrument used in this study to classify individuals according to their psychological type was the Myers-Briggs Type Indicator (1962). The Myers-Briggs Type Indicator (MBTI) was designed explicitly to operationalize Jung's psychological types theory. Essentially, it is a forced choice, self report inventory. The instrument provides four separate indices for each of the four basic preferences that, under Jung's theory, structure the individual's personality.

Although analysis of data produced by the study did not offer conclusive findings, the direction of the evidence supports the following inferences regarding CMI users:

- 1. Apprehension of human-machine systems is associated with feeling, as a mode of judgment, in preference to thinking.
- 2. Confidence in the system is associated with a thinking preference.
- 3. System usefulness is associated with users who had sensation and thinking preferences.

An intrinsic motivation for pursuing this research was the concern for the lack of information related to the interface between users of a decision support system and the system itself. It has been argued that research related to the psychology of MIS users is a key variable that has been disregarded in research efforts related to technological innovations and their implementation (Mason & Mitroff, 1973). The thesis of this study, therefore, was that the design CMI system should incorporate the personality or psychological typology of the users into the overall design.

The findings of this study indicated that a significant portion of the wide variance in the success of the CMI system

implementation is attributable to psychological type. Contributing to such success or failure were such variables as confidence in the system and its perceived usefulness. Stated another way, information that was useful to one person was not useful to another.

If the thesis of this research is judged to be important and if its general conceptual framework and design are considered valid, there are several considerations that may be emphasized in future research and development.

- 1. A more complete understanding of the implication of psychological type in decision making is necessary in order to refine and enhance systems of inquiry and information generation.
- 2. Alternative methods of supplying information to the instructional decision maker should be explored in concert with research related to psychological type. This may involve structured experimentation with different modes of information presentation.
- 3. Research regarding appropriate methods of evidence generation is needed for the design of future management information systems. Evidence generators that are appropriate for a person of one psychological type may not be appropriate for another.

Problem

Problems faced by people concern the nature and structure of the question or proposition for which a solution is sought or about which a decision must be determined. An interesting classificatory method for problem and problem environments is offered by Harrison (1977). This classification is based on (a) the degree and nature of human involvement and (b) the structure of the system. Such a model presents six possibilities.

	Non-People	Semi-People	People
Well-Defined (Deterministic or Mildly Stochastic)			
Ill-Defined (Unstructured or Uncertain)			

Problems and decisions for which a CMI/IMIS system are designed are ill-defined, people systems. This classification represents the most difficult set of conditions with respect to instrumentation and operationalization. The application of technology as a process of analysis and design represents an area which has received insufficient attention

112 in the MS/OR literature. Failure to correctly address this variable may lead to four modeling errors identified by Sutherland (1973): reductionism, improper analogies, improper assumptions, and expediency.

Organizational Context

The organizational context in which the decision maker functions and in which the system will be implemented may significantly affect the overall design format. For example, a CMI/IMIS designed for use in an organizational structure such as Individually Guided Education (IGE) may be totally inappropriate for other instructional programs. As in the case of the person variable, information and information systems which are suitable for one context may not be at all suitable for another.

Evidence

Evidence, the fourth variable in the Mason and Mitroff framework, represents the information upon which the decision maker will reach a solution or formulate a strategy. In a CMI/IMIS, algorithms or other analytical methods are employed in the processing of raw data. These must be consistent with the precepts of the instructional program and curricular design. The user will have a greater appreciation of the evidence if he/she is involved in the design of the evidence generators.

Mode of Presentation

The last variable, mode of presentation, may be considered so trivial that its importance within the total system design and its impact on the success of system implementation is underestimated--if not totally neglected. Many CMI/IMIS designers are engaged in computer science activities and may, therefore, reflect their own personal bias and preference in the way system information is presented. As a result, it is often presumed that output is best presented in the form of computer printouts or display on a terminal. Such a format is obviously not the only alternative in many applications. Certain users (of varying psychological type) may find different modes of presentation and other representations of information to be of considerably greater use.

For example, text or narrative interpretations of statistical data which capture the meaning of the data may be more readily received by certain persons. Scenarios can often be used to convey the impact of manipulating certain parameters. Inclusion of only the most important information, screening out ancillary output, may be a productive

technique. Media specialists can provide expertise with respect to effective strategies and audio-visual techniques.

Conclusions

In the opinion of the author, CMI/IMIS represent important potential for the enhancement of student learning. A recent review of operational CMI systems indicates numerous efforts have offered proof-of-concept.² Unfortunately the success record of implementations is not good. This paper has discussed several factors which may be associated with problems and failures of systems.

The need for such an instructional management device clearly exists and the engineering technology is available. Additional investigation is, however, needed in the domain of design processes, particularly as they impact on the implementation. A growing body of evidence exists which demonstrates that successful implementation and utilization are directly related to the meaningful involvement of users in the planning and designing processes.³ CMI/IMIS design, implementation, utilization, and evaluation must occur within a total systems framework with attention given to all the areas discussed herein. Failure to do so will increase the likelihood of adding one more system to the growing list of unsuccessful efforts.

Footnotes

This research was supported in part by funds from the National Institution of Education, Department of Health, Education and Welfare to the Wisconsin Research and Development Center for Cognitive Learning under Contract No. NE-C-00-3-0065. The opinions expressed herein do not necessarily reflect the position or policy of the National Institute of Education and no official endorsement by that agency should be inferred.

1. A complete discussion of this research may be found in Technical Report No. 454, Wisconsin Research and Development Center for Individualize Schooling, Madison, WI 53706.

2. This conclusion is drawn from a report prepared by the author while serving as consultant to the Alaska Telecommunications Project, a project of the Alaska State Department of Education and the Northwest Regional Educational Laboratory. A review of this report will appear in AEDS Journal (Summer 1979, 12).

3. For additional information and research related to this topic, the

reader is referred to Professor Gerald Nadler, Department of Industrial Engineering, University of Wisconsin-Madison.

References

- Bozeman, W. C. Human factors considerations in the design of systems of computer managed instruction. AEDS Journal, 1978, 11, 89-96.
- Harrison, H. L. Systems analysis for social-science professionals. Unpublished manuscript, 1977. (Available from the University of Wisconsin-Madison, Department of Mechanical Engineering).
- Jung, C. G. Psychological types. New York: Harcourt, Bruce, & Co., 1923.
- Mason, R. O. & Mitroff, I. I. A program for research on management information systems. Management science, 1973, 19, 473-487.
- Myers, I. B. Myers-Briggs type indicator manual. Palo Alto: Consulting Psychologists Press, Inc., 1962.
- Sutherland, J. W. A general systems philosophy for the social and behavioral sciences. New York: George Braziller, 1973.
- Thomas, David B. The effectiveness of computer-assisted instruction. AEDS Journal (to appear in Summer 1979, 12).

THE INTERACTION OF HEURISTIC/ALGORITHMIC
INSTRUCTIONS AND CONCEPTUAL PREFERENCE
IN COMPUTER BASED PROBLEM SOLVING

Joan M. Sustik
Bobby R. Brown

Computer Assisted Instruction Laboratory
University of Iowa

Undergraduates varying in conceptual preference, measured by the Siegel and Siegel Educational Set Scale, received heuristic or algorithmic problem solving training. Subjects solved a series of computer based mathematical problems using a multiple choice or numerical list response mode. Regression analysis revealed disordinal interaction wherein the Heuristic-List treatment was negatively related, and the Algorithmic-Multiple Choice treatment was positively related to conceptual preference. Performance was stable across the conceptual level continuum for students in the Heuristic-Multiple Choice and the Algorithmic List treatments. Results suggest that conceptual preference is impacted more by degree of structure imposed in responding than by structure in problem solving training.

INTRODUCTION

Computer Assisted Instruction (CAI) will realize its potential in adaptive instruction to the extent that appropriate instructional contingencies can be identified and included in the instructional designer's repertoire of strategies. A basic paradigm used to derive information regarding these instructional decisions is one known as "Trait-Treatment-Task-Interaction" (TTTI). Use of this paradigm as a model for research acknowledges, and indeed is based on, the assumption that the optimal "treatment", that is, the optimal nature of the CAI program, is best identified in joint consideration of the particular task to be accomplished and of specific learner characteristics relevant to these tasks (Berliner and Cahen, 1973; Messick, 1976; Divesta, 1974, 1975; Cronbach and Snow, 1977).

CAI is inherently suitable for implementing instructional contingencies using response tailored instruction based on TTTI findings both prior to and during instruction, that is, at both macroadaptive and microadaptive levels. While it would certainly be desirable to use CAI's ability to implement complex instructional algorithms at the microadaptive level, it is important and necessary to initially identify the elements of successful adapting at the macro level.

How then, should identifying instructional contingencies at the macro level proceed? The logical and recommended way is to specify the particular learning task to be accomplished. From there, learner characteristics which, theoretically at least, are required for successful task completion should be identified. This is followed by designing different instructional treatments which should, again theoretically, facilitate task completion such as problem solving for students with different characteristics.

Problem solving skills are pervasive in life, and in academic pursuits. In particular, such skills may be maximally exercised through CAI. As a result, we at the Computer Assisted Instruction Lab at The University of Iowa have been interested in factors which influence computer based problem solving success. For this study then, the task identified occurred in the problem solving domain of instructional outcomes.

RATIONALE AND REVIEW OF LITERATURE

This study investigated learner and instructional treatment variables postulated to interact in a problem solving task. The results were to be used for the purpose of aiding instructional design decisions inherent in computer based individualized instruction, as based on the TTTI paradigm.

The task used the Luchin's Water Jar Test (WJT). This test is a series of mathematical problems designed to assess mechanization in method of response (Luchins, 1942, 1959). Students are encouraged to develop a consistent solution strategy in the beginning of the problem series with the "set" problems as shown on Table 1. Subsequent problems, referred to as "criticals," can be solved using the initial strategy, or using a more direct or more efficient strategy. Ultimately, a new strategy must be employed to solve the final problems in the series, known as "extinction" problems. A problem solver who persists in using the initial method when it is no longer most efficient or no longer viable is said to be responding in a mechanized manner.

Table 1
Problem Series Used

#	PROBLEM	POSSIBLE SOLUTIONS	TYPE OF PROBLEM
1	11-25-5 get 4	B-A-2C	SET
2	21-127-3 get 100	B-A-2C	SET
3	14-163-25 get 99	B-A-2C	SET
4	18-43-10 get 5	B-A-2C	SET
5	9-42-6 get 21	B-A-2C	SET
6	20-59-4 get 31	B-A-2C	SET
7	23-49-2 get 20	B-A-2C A-C	CRITICAL
8	15-39-3 get 18	B-A-2C A-C	CRITICAL
9	28-76-3 get 25	A-C	EXTINCTION
10	40-118-18 get 62	2A-C	EXTINCTION

In analyzing the task, it was postulated that learners most susceptible to response mechanization would be those least inclined to generate and test additional hypotheses on method of solution. The hypotheses generating dimension is conceptually similar to an individual difference factor which Gagne (1966) identified as operating in problem solving. Gagne indicated that "fluency in making new combinations," which affects the internal processes of combining subordinate rules and devising a provisional rule, is an individual difference factor which could be influenced by the instructional factor of guiding thinking.

The cognitive style of cognitive simplicity-complexity was identified as the learner variable pertinent to the task at hand. Cognitively complex individuals possess more sophisticated conceptual systems, and are more attuned to diversity, conflict and uncertainty such as exists in the WJT where new solution methods are required. This dimension, investigated by Tomlinson and Hunt (1971) using the paragraph completion test (PCT), was found to be influenced by instructional methods. Low Conceptual Level students benefited most from a rule-example order of instruction and High Conceptual Level students were amenable to either inductive or deductive methods. Gordon (1977,a,b) investigated the relationship between conceptual level and the problem solving process and found that High Conceptual Level students provided more math examples prior to discussion of the rule in teaching geometry, and chose a "Polya" heuristic method of teaching. Low Conceptual Level students used fewer examples prior to the rule and chose a "Dahlmus" method which emphasized translating each verbal statement into a specific mathematical statement without necessarily considering the problem as a whole. Gordon suggested that matching problem solving training method to cognitive style, such as the Polya method to HCL, was an area which warranted further examination.

The cognitive complexity dimension has also been investigated using the Educational Set Scale (ESS), a forced choice objective measure by Siegel and Siegel (1965) to assess a student's preference for factual or conceptual material.

Siegel referred to this dimension as "educational set", a continuum ranging from extremely factually oriented, which can be interpreted as indicating cognitive simplicity, to extremely conceptually oriented, indicating cognitive complexity. A factually set person is one who prefers to deal with discrete facts, adding units of information to his cognitive structure without relating them, while a conceptually set person deals with facts by either ignoring them or incorporating them into existing systems. Most people tend to fall near the middle of this continuum but for those on the extreme, possession of either "educational set" may result in differing reactions to courses, materials, methodology or instructors. In a televised college class, Siegel and Siegel found that conceptually set students learned both concepts and facts better than factually set students. Performance by conceptually set students was adversely affected when factual material was reinforced; the same relation held for factually set students when conceptual material was reinforced.

The present study sought to train students to solve problems using either an algorithmic or heuristic approach and to determine whether this approach would be differentially effective for students who vary in cognitive complexity as measured by the ESS. Heuristic methods of problem solving teach the learner to develop a general process of problem solving while algorithmic methods teach a set of rules which can be used to solve a specific set of problems. Specific response modes used in CAI may interact to facilitate or hinder either heuristic or algorithmic instruction. For example, the multiple choice response mode, wherein the student may select a correct solution equation from a list of possible solution equations, can be conceived of as being more closely aligned to the wholist heuristic approach. Similarly, a numerical list, which requires specific stepwise completion of a given problem, can be interpreted as being more amenable to the algorithmic training method. Therefore, the study also investigated what impact two commonly employed computer based response modes, multiple choice or numerical list, would have on these two problem solving methods. The multiple choice mode was conceived as being matched in degree of structure with the heuristic method, while the numerical list mode, in forcing stepwise completion, was viewed as structurally matched to the algorithmic training. Factually set students theoretically, would prefer the structured algorithmic approach while conceptually set students should prefer the heuristic approach. In a prior study by Sustik and Brown (1978) investigating the impact of response modes on solving the WJT, large variances within response mode groups resulted which suggested the presence of an individual difference dimension operating to increase the variances.

Conceptualized in the TTTI framework this study investigated the interaction between one trait variable (cognitive simplicity-complexity), two treatment variable (Heuristic or algorithmic training and numerical list or multiple choice response mode) in a math problem solving task. The four combinations of method and mode were Heuristic-Multiple Choice (HMC), Heuristic List (HLIS), Algorithmic-Multiple Choice (AMC), and

Algorithmic-List (ALIS). The dependent measures of interest were method of solution, total time in problem and, for the list response mode groups, total number of steps in problem, all measured on the critical and extinction problems. Solution methods could be categorized as "no solution", or as "mechanized" if they were repetitions of previously used methods, or as "non-mechanized" if they were more efficient than the repeated method. Methods were accordingly assigned a "0", "1" or "2" with lower scores thus indicating less successful and more mechanized problem solving performance.

METHODS

One hundred twenty nine upper level students enrolled in a basic media course at the University of Iowa participating on a voluntary basis in Fall 1977. Participation consisted of an online: introduction to CAI, problem solving training and the WJT series. Two weeks prior to participation, the ESS was administered to students and students were sequentially assigned to one of four treatments on the basis of their ESS ranks: Heuristic Multiple Choice (HMC), Heuristic List (HLIS), Algorithmic Multiple Choice (AMC), and Algorithmic List (ALIS). Data was analyzed for all students and, in keeping with established literature, on only those students who solved the last two set problems, referred to as the "Luchin's Standard". Regression analyses were conducted to assess the interaction of ESS and instructional treatment on method of solution, total time in problem and, for list groups, number of steps in problem. Where no interactions resulted, group differences were analyzed.

RESULTS

No interactions resulted when data from all students was analyzed. Analysis on Luchin's Standard revealed the following disordinal interactions. The AMC facilitated non-mechanized performance for conceptually set students while it had a debilitating effect on factually set students. The opposite relationship held for HLIS group, not only for method, but for time and steps as well. Variation in ESS score was not accompanied by variation in performance for students in the HMC or ALIS groups.

In considering all data, the multiple choice response mode groups exhibited less mechanized response and spent less time in problem than did numerical list response mode groups. Within response modes, the algorithmic training yielded higher performance than heuristic training. Across subjects participants spent a longer time in item 10, and were less successful in solving it, suggesting that while items 9 and 10 were both "extinction" problems, they were not of equivalent difficulty.

Table 2

Regression Summary Table for Interaction
Followup Comparisons for Method
of Solution, Luchin's Standard

	n	R ² Full Model ^a	R ² Reduced Model ^b	F
Method 7-10				
AMC-HLIS	(48)	.35317	.28152	5.095*
HMC-AMC	(54)	.48696	.39841	8.975**
HMC-ALIS	(49)	.22026	.22026	-
HLIS-ALIS	(47)	.18555	.07496	6.110*
HLIS-AMC	(52)	.51553	.33379	18.7566**
AMC-ALIS	(53)	.43491	.37553	5.359*

Note: a R² Full Model = R²y·ESS, D1, XD1

b R² Reduced Model = R²y·ESS, D1

* = p < .05

** = p < .01

Table 3

Regression Summary Table for Interaction
Followup Comparisons for Total Time
in Problem, Luchin's Standard

	n	R ² Full Model	R ² Reduced Model	F
Total time 7-10				
HMC-HLIS	(48)	.67371	.61884	7.399**
HMC-AMC	(54)	.28426	.23865	3.186*
HMC-ALIS	(49)	.62918	.62918	-
HLIS-ALIS	(47)	.37451	.10431	18.575**
HLIS-AMC	(52)	.67514	.67190	.478
AMC-ALIS	(53)	.70375	.66910	5.7295*

Note: a R² Full Model = R²y·ESS, D1, XD1

b R² Reduced Model = R²y·ESS, D1

* = p < .05

** = p < .01

Table 4

Regression Summary Table for Overall Interaction
on Total Number of Steps in Problem for
HLIS - ALIS, Luchin's Standard

	R ² Full Model ^a	R ² Reduced Model ^b	F ^c
Step 10	.32241	.11585	13.178**
Steps 7-8	.25143	.20114	3.02316*
Steps 9-10	.37944	.15688	16.139**
Steps 7-10	.38740	.17606	15.523**

Note: n = 47

a R² Full Model = R²y·ESS, D1, XD1

b R² Reduced Model = R²y·ESS, D1

c df = 1.43

* = p < .05

** = p < .01

Adjusted Means on Method of Solution for All Students*

	<u>$\bar{X}$ 10, adj.</u>	<u>S. D. 10</u>
HMC	1.2985	.973
HLIS	.398417	.803
AMC	1.76584	.654
ALIS	.676867	.970
	<u>$\bar{X}$ 9-10, adj.</u>	<u>S. D. 9-10</u>
HMC	3.28986	.973
HLIS	1.89748	1.348
AMC	3.64776	.917
ALIS	2.60138	1.098
	<u>$\bar{X}$ 7-10, adj.</u>	<u>S. D. 7-10</u>
HMC	6.19356	1.078
HLIS	4.58312	2.466
AMC	6.53143	1.419
ALIS	5.61042	1.672

*Values Assigned to Method:

"0" - no solution

"1" - mechanized

"2" - direct

DISCUSSION

In discussing the impact of the instructional treatments on performance, it is appropriate to consider the nature of heuristic and algorithmic instructions, as well as the numerical list and multiple choice response modes. The heuristic problem solving process relied on identifying components of the problem, stimulating recall of past similar problems where a method or parts of the method could be employed in the current context, and devising a plan pertinent to the problem at hand. This approach guides the process for developing a problem solving program, but does not guarantee success. It is a relatively unstructured problem solving approach, stimulating production of mental alternatives from which to build a strategy for solution.

Algorithmic problem solving process, by contrast, called for developing a more focused, highly structured series of steps that, once useful, could be employed in subsequent similar problems. In brief, the heuristic process suggested development of a plan for each problem, while the algorithmic process called for reconstituting a mechanical procedure which worked previously.

The multiple choice response mode presented several solution equations. In terms of structure, it was more compatible with the heuristic method, since each solution equation could be conceived of as representing a plan or program for solution. The numerical list, by contrast, in forcing students to proceed in a specific stepwise manner, could be conceived of as being more

compatible with the algorithmic approach which stressed steps in a solution.

In terms of the structuring imposed on the students, then, the instructional treatments represented the following combinations:

<u>TREATMENT</u>	<u>PROBLEM SOLVING</u>	<u>RESPONSE MODE</u>
HMC	low	low
HLIS	low	high
AMC	high	low
ALIS	high	high

In the ALIS and HMC treatments, there was little variation in method of solution or total time in problem along the ESS continuum. It appears that a match in the degree of structure elicited in the problem solving training and the degree of structure imposed by the response mode does not result in differential performance by factually or conceptually oriented students.

Disordinal interactions on the method of solution occurred for the HLIS and AMC treatments where there was a mismatch between the problem solving training and response mode structure. Conceptually oriented students were less successful in problem solving under the HLIS treatment while factually oriented students performed well under this treatment. This may possibly be due to the fact that conceptually oriented students, predisposed to a low degree of structure, are primed to think in an unstructured mode under the heuristic treatment and then required to respond in a highly structured stepwise manner. This may have motivationally impeded their performance. Or it may be that while they had the desire to

solve the problems, they were not able to solve the problem using the list response mode. However, conceptually oriented students in the HLIS treatment also spent less total time in problem, and took fewer steps in the problem solving process than did the factually oriented students. This supports the possibility that high ESS students in this mode were less motivated to complete the problem, opting for the no solution option earlier than their low ESS counterparts.

Finally, while the WJT share similarities with problem solving in general, in discussing student performance it is necessary to bear in mind that the tests constitute a specific subset of problem solving situations. The primary interest was in mechanization of solution method in a mathematical problem solving series. This concern was whether using a single viable method in a series of problems would cause the learner to perseverate on this method, precluding developing alternative methods in subsequent problems which require different methods to achieve solution.

This is not to imply that the concept of mechanization is without real world correlates. Consider, for example, a physician who consistently employs a series of diagnostic tests to identify the cause of certain symptoms. In a situation analogous to the critical items in the WJT, there may be fewer, less costly, and less time consuming diagnostic tests which the physician could order - here, non-mechanized diagnosis would be preferred. Mechanization in extinction-type problems would, of course, be universally undesirable, since prior methods would negate deriving new techniques. The present study, however, used math problems established in the literature to assess mechanization.

CONCLUSION

In general, results indicate that compatibility in degree of structure elicited in problem solving training and that imposed by the response mode leads to performance which is stable across the conceptual level continuum. A mismatch in the degree of structure yields differential performance for students varying in conceptual level. Conceptually set students profit from the unstructured response mode (AMC) while factually oriented students profit from the structured response mode (HLIS).

The findings suggest avenues for adaption in computer based problem solving programs. The ESS has been administered via CRT with correlation of .895 ($n=17$) on paper pencil version. Appending the ESS program to problem solving programs would provide an immediate and invisible means of individualizing via branching to HLIS or AMC type treatment on the basis of the ESS score.

Finally, it is perhaps unlikely that the recommendations of this or similar studies, will be immediately implemented. Perhaps the greater significance lies in validating the occurrence of TTT interactions in computer based instruction. This suggests the need to not only be aware of

their possible existence in assessing CAI outcomes, but also the need to identify those which are most prevalent and which can lead to greatest improvement in designing instruction. The extent to which that interaction identified here is significant remains to be seen with repetition over time.

REFERENCES

- Berliner, D.C. and Cahen, L.S. Trait Treatment Interaction and Learning. In F.M. Kerlinger (Ed.) Review of Research in Education, Vol. 1, Itasca, Illinois Peacock Press, 1973.
- Cronbach, L. and Snow, R.E. Aptitudes and Instructional Methods, New York: Irvington Publishers, 1977.
- DiVesta, F. Theories and Measures of Individual Differences in Studies of Trait and Treatment Interactions. Educational Psychologist, 1973, 10(2), 67-75.
- DiVesta, F. Trait Treatment Interactions, Cognitive Processes and Research on Communication Media. Audio Visual Communication Review, 1975, 23(2), 185-197.
- Gagne, R.M. Human Problem Solving Internal and External Events. In B. Kleinmütz (Ed.), Problem Solving Research, Theory and Method, New York: John Wiley and Sons, 1966.
- Gordon, M. Mathematics Presentation as a Function of Cognitive Personality Variables. Journal for Research in Mathematics Education, May, 1977.
- Gordon, M. Perspectives in Problem Solving: Person Vs. Paradigm. Paper presented at the American Educational Research Association, New York, 1977.
- Luchins, A.S. Mechanization in Problem Solving. Psychological Monographs. 1942, 54(6).
- Luchins, A.S. Rigidity of Behavior. Eugene, Oregon: University of Oregon Books, 1959.
- Messick, S. Personal Styles and Educational Options. In S. Messick and Associates, Individuality in Learning, San Francisco, Jossey-Bass Publishers, 1978.
- Siegel, L. and Siegel, L.C. Educational Set: A Determination of Acquisition. Journal of Educational Psychology, 1965, 56(1), 1-12.
- Tomlinson, P. and Hunt, D.E. Differential Effects of Rule-Example Order as a Function of Learner Conceptual Level. Canadian Journal of Behavioral Science, 1971, 3(3), 237-246.

INTERACTIVE COMPUTER-BASED EXERCISES IN RADIOLOGY

Jeffrey Kreps, Robert Kraft, Russell Burris
University of Minnesota

and

Frederick Olson, M.D.
St. Paul-Ramsey Hospital

SQUIRE is a computer-based system of exercises in diagnostic radiology intended to be used as a supplement to current methods of instruction for third and fourth year medical students. The students view actual X-ray films while SQUIRE conducts a Socratic dialogue on the films. The dialogue was modeled after radiologist-student interaction and is designed to simulate this type of interaction. Direct comparison of films is stressed to enhance perceptual learning. A model of the expert diagnostic process guides the overall sequence of questions while a set of Socratic production rules determines the specific questions and feedback provided to the students for a given film set. SQUIRE may be used for research purposes to track student learning and to collect data to compare expert radiologist, physician, and student performance.

Radiology is a difficult area for third and fourth year medical students to master. Reading and interpreting the various lines and shadows that make up an X-ray film is a complex perceptual and diagnostic skill. Yet it is important for medical students to learn a great deal about radiology before they become physicians, so that they understand how to make use of various radiological techniques for the benefit of their own patients. Better understanding of radiology may also enhance the overall quality of medical training because this knowledge can be used to correlate and retain the innumerable details encountered in medical school (Squire and Becker, 1975).

SQUIRE is an interactive computer-based system of exercises to be used by medical students in a course in radiology. SQUIRE is intended to be used as a supplement to texts, lectures, and other instructional materials currently used in most radiology courses. These computer-based exercises are designed to provide structured practice in radiological diagnosis (reading X-ray films) in a realistic task (teaching) environment. The exercises use actual radiographs (X-ray films) which the students examine. SQUIRE then engages the students in a Socratic dialogue, asking questions and providing feedback about the films they have in front of them. The dialogue is designed to sound as natural as possible and to simulate the kind of Socratic interaction provided by an experienced radiology teacher. SQUIRE attempts to provide the type of interaction which staff radiologists, given enough time, would like to be able to

engage in with small groups of students, but which in reality, is seldom possible. Practice in applying knowledge in a realistic task environment is an important factor in the acquisition of problem solving skills which has been shown to hold true regardless of whether the skill is in chess (deGroot, 1965, Chase and Simon, 1973) or medical problem solving (Elstein, Schulman, and Sprafka, 1978). SQUIRE is designed to provide this type of applied practice with real X-ray films.

Instruction in most areas of medicine typically provides medical students with structured practice in diagnostic reasoning; however, instruction in radiology must go beyond simply building up diagnostic reasoning powers in order to be effective. Instruction in radiology must concern itself with teaching the students how to see with a critical eye--you cannot diagnose what you do not see. So, training in perceptual learning is essential because it represents the first important step in the process of learning how to interpret X-ray films. Also, a large number of perceptual errors are made by practicing physicians as they interpret X-ray films. Garland (1959) and Tuddenham (1962) have pointed out that experienced physicians typically miss 25 to 30 percent of abnormal chest radiographs under conditions in which these errors are primarily due to failures of perception.

So, what is this thing called perceptual learning? In general, perceptual learning refers to an increase in the ability to extract information from the environment as a result of experience and practice (Gibson, 1969, p. 3). This process is not

120 passive absorption, but rather, it is an active process (Gibson, 1966). One important visual skill which is similar to the radiologist's skill involves the interpretation of aerial photographs. It takes a great deal of practice to identify a variety of different landmarks from the rather uniform gray field of an aerial photograph. The Russian missile bases installed in Cuba in 1962 were discovered by one perceptually-educated man's detection of them in a photograph (Gibson, 1969, p. 8). The radiologist's perceptual task is similar, yet more complex. So, what is involved in "perceiving as a radiologist?"

Perceptual learning in radiology can be separated into three problems. The first involves developing the ability simply to detect an abnormality in an X-ray film; or, to put it another way, the first problem involves distinguishing an abnormal X-ray film from a normal one. We observed that after only three weeks of training, students actually can locate and describe abnormalities in an X-ray film that they could not even see prior to their radiology training.

The second important radiological problem confronting the perceptual learner involves learning efficient search techniques--learning where to look. Tuddenham (1962) asserts that perceptual errors arise from incomplete search of the radiograph either because of bad scanning habits or because the identification of one lesion leads to premature termination of the search procedure. Tuddenham (1962) cites several examples of physicians accurately identifying one lesion, abandoning search, and missing a second lesion. Further, accurate perception of critical roentgen shadows does not depend solely on the physical quality of the radiograph. The findings which are overlooked are often gross and perfectly obvious, at least in retrospect. In Tuddenham's study, the most commonly missed finding was the amputation of a female breast which should be detected--given efficient search habits. SQUIRE lays the foundation for accurate and efficient perceptual search by structuring the way in which the students examine the X-ray films in accordance with a general model of the radiological search process.

Thirdly, perceptual learning in radiology involves the abstraction of invariant features within a certain pathology and the detection of feature differences among different pathologies. This appears to be the problem area which presents the most challenge--and mystery. It is, however, the area which we believe SQUIRE will help us describe with more precision.

In general, when learning to identify and categorize a set of visual stimuli, the learner must be taught to perceive feature differences between different stimuli through direct comparison of the stimuli. When first-grade instructors

teach the letters of the alphabet, they group together visually similar letters and teach the students the distinguishing features. For example, they present the letters, "b, d, q, and p," not "m, b, x, and l," which allows the students to pick up just the necessary information needed to distinguish between different letters of the alphabet. Similarly, when learning to read X-ray films, the medical student must learn to detect invariant features across different examples of the same pathology--for example, the student should learn to detect pneumonia when it is in the upper right lung field as well as when it is in the lower left lung field; and the medical student must learn to detect the critical signs that differentiate perceptually similar examples of different pathologies--for example, the student should learn to see the difference between cancer and pneumonia even when both present themselves as fuzzy white patches in the lower right lung area.

SQUIRE encourages the detection of invariant relations and distinctive features by having each exercise involve four X-ray films which the students must simultaneously compare and contrast. Such direct confrontation between several films allows the students to see how a single disease may present itself in a variety of ways and also allows the students to pick up directly those signs which distinguish two different pathologies.

Given that there seem to be some problems with current methods in radiology instruction, what are some possible ways in which the teaching of radiology may be improved? A model for what we consider to be effective teaching of radiology is the course and materials developed by Dr. Lucy Squire at the Downstate Medical Center in New York. Several important elements in our computer-based exercises are outgrowths of effective techniques which Dr. Squire has developed through many years of experience in teaching radiology to medical students. For example, Dr. Squire and her colleagues found that a student working alone on programmed materials actually learns less than when he works in a group with several other students (Squire, Blotnick, and Becker, 1972). They theorize that the students succeed in learning because they are able to pool their background reading, clinical experience, and problem solving skills. In fact, Squire, et. al. (1972) observed that a group of fourth-year medical students using their collective knowledge and experience out-performed a first-year radiology resident working alone with unfamiliar material. Similarly, Burris, at the University of Minnesota, has reported that the computerized law exercises are more effective when several students work together on them.

Based on this evidence, our computer

exercises are designed to be utilized by groups of students rather than individuals. SQUIRE's questions are intended to encourage discussion; indeed, at critical points in the dialogue, the program specifically asks the students to discuss their answers before responding.

Another idea which we have derived from Dr. Squire's experience is that students' should view sets of radiographs rather than a single film case (Squire, Twersky, Pals, and Becker, 1975). They state that the dramatic impact of matched sets of films helps students remember what they learn, and that comprehension is enhanced by contrast. We can take this explanation farther by considering the process of perceptual learning. Sets of films enhance perceptual learning by emphasizing invariant features across several examples of one disease and contrasts between different diseases. The Socratic questions posed by our program focus on the critical features of each film as they relate to the features of the other films in the same set. By focusing on these critical features and by stressing direct comparison between films, SQUIRE's questions aim directly at the perceptual learning process, i.e., teaching students to "see like radiologists."

Considerations in the development of the format and structure of SQUIRE's dialogue came from several other sources. We found several radiology texts (Felson, 1965; Squire, 1975; Paul and Juhl, 1972) to be helpful, but our primary source for the Socratic questions which the program asks, was our observations of radiologist-student interactions and from later interviews with the radiologists about the questions they asked. Additional ideas for questions came from Allen Collins' theory of Socratic dialogues (Collins, 1977).

Program Structure

In view of the fact that SQUIRE's questions and feedback are intended to induce both efficient, accurate scanning techniques and appropriate diagnostic reasoning, we first formulated a model describing the X-ray film reading process. Our plan was to organize the sequence and structure of the Socratic dialogue according to this process model. The notion of structuring the teaching process so that it corresponds to a description of the process by which an expert radiologist scans and interprets X-ray films makes intuitive sense when one recognizes that an important goal of radiology instruction should be one of teaching students to "think as radiologists." We collected verbal protocols (Newell and Simon, 1972) from several radiologists as they scanned and interpreted X-ray films. We encouraged them to "think out loud" as they read the films. Based on these observations of expert radiologists and on a careful study of radiology texts, we developed the model

describing the process of diagnosis in radiology shown in Figure 1. This model specifies the overall sequence of SQUIRE's questions and is designed to lead students through the reading of an X-ray film in a systematic way.

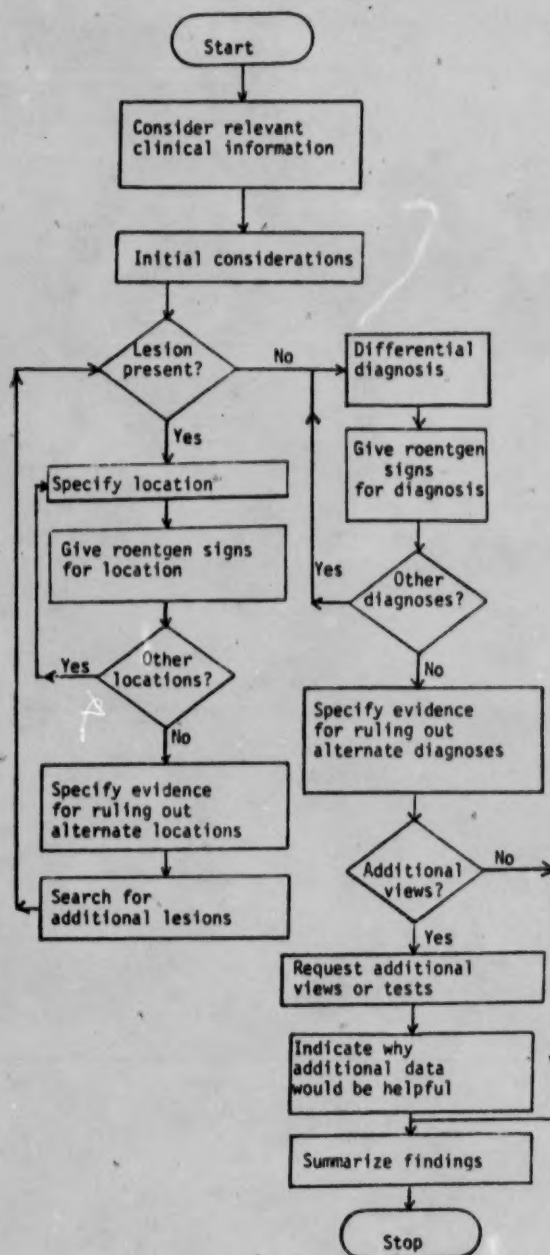


Figure 1. The process of reading a radiograph.

As one looks at the model in Figure 1 it is clear that reading a radiograph is a systematic process. Although all radiologists do not read X-ray films the same way, each radiologist adopts some system (a "personal equation") to insure that all portions of the radiograph are examined. This system increases the probability that all abnormalities will be detected (even if a large lesion predominates the visual field). We observed that prior to searching for abnormalities, the expert radiologist considers a variety of factors which may influence the reading and interpretation of the radiograph. These "initial considerations" are the first elements of the process represented in Figure 1. For example, what is normal for a woman may be quite abnormal on the radiograph of a man. So SQUIRE sometimes asks, "What sex do you think the patient is?" Similarly, what is normal for a person of 60 may not be normal for a teenager, so when appropriate, the program asks, "Do you think the patient is young, middle aged, or old?" Because the view, technique, and position of the patient when the film is taken dramatically alter the appearance of the bones and organs, the program sometimes asks, "What view do you think this is?" or "Is there anything wrong with the technique for this film?" Finally, because certain abnormalities provide clues as to the nature and location of other abnormalities, e.g., a missing rib may indicate previous heart surgery, and a missing breast should be identified because it indicates prior history of cancer, SQUIRE may ask, "Do you see any abnormalities in the bones or soft tissues on this film?"

Once the radiologist has systematically considered these initial factors, the program moves on to perhaps the most important phase in the diagnostic process: detecting and determining the location of any abnormal signs on the film. Corresponding to this phase of diagnosis, SQUIRE asks, "Where do you think the lesion is located?" and then requires the students to provide evidence for their decision by asking, "What signs do you see which indicate the lesion is located there?"

Following the students' responses to these two questions, SQUIRE provides feedback not only in terms of whether the answers were correct or incorrect for the film being studied, but feedback which explains how and why the students' reasoning was incorrect if in fact it was. The program can be especially effective (perhaps surpassed only by the human teacher) at providing feedback aimed directly at specific perceptual and logical errors. In the next section we will examine the program features which enable SQUIRE to provide this kind of contingent feedback. Continuing through Figure 1, we find "Differential diagnosis" following the location phase in the flowchart. The

questions asked by SQUIRE corresponding to this diagnosis phase are similar to the questions dealing with location. The program asks, "What is your initial diagnosis?" and "What evidence do you have to support your choice?"

In both the location and diagnosis phases of our flowchart there are decision points where "other locations" or "other diagnoses" are considered by radiologists. Elstein, et. al. (1978) stress the importance of training medical students to generate multiple competing hypotheses as the basis for improving their diagnostic skills. SQUIRE encourages this method of problem solving by requiring the students to consider several possibilities for both location and diagnosis. Immediately following the students' initial choice for location, the program will ask, "Do you think there is another possible location?" and after their diagnosis, "Do you think there is evidence for other possible diagnoses?" The questioning for a given film concludes by asking the students about additional views or tests which they think would provide useful information for locating or diagnosing the abnormalities on the film.

One of the most important features of SQUIRE is that all the individual dialogues for each set of radiographs are generated from a single program. After a radiologist selects an X-ray film for inclusion in a problem set, he then calls up SQUIRE's author program. The author program poses questions identical to those asked by the instructional program. The radiologist's answers are then stored in list structures, and it is the elements from these lists which SQUIRE uses to generate the specific questions and feedback in the dialogue with the students.

An example of the lists with which the program generates the portion of the dialogue dealing with "location of the lesion" is shown in Figure 2. The radiograph selected for this example is a right middle lobe pneumonia. Under "location of lesion" for this film (list #1 in Figure 2) the radiologist has provided "Right Middle Lobe." The signs which indicate that the lesion is located in the right middle lobe are "Right Heart Border Obscured" and "Air Bronchogram Sign." These reasons in list #2 are linked to the locations in the first list. The location of a pneumonia for another film in this set, this time a right lower lobe pneumonia, is also stored in list #1, and the sign which indicates "Right Lower Lobe," namely "Right Diaphragm Obscured" is stored in list #2.

The following dialogue is a sample of computer-student interaction which SQUIRE can generate from the items in lists 1 and 2. In this example the students will correctly identify the abnormal sign--right heart border obscured--but will incorrectly reason the location of the lesion.

List #1	List #2	List #3	List #4
Location of Lesion	Evidence for Location	Alternative Locations	Reason Ruled Out
Right Middle Lobe	Right Heart Border Obscured	Right Lower Lobe	Right Diaphragm Intact
	Air Bronchogram Sign	Right Pleural Space	Air Bronchogram Sign
Right Lower Lobe	Right Diaphragm Obscured		

Figure 2. An example of lists used by SQUIRE to generate the portion of dialogue dealing with "Location of Lesion."

Note that the underlined phrases in the dialogue are items from the lists.

SQUIRE: Where do you think the lesion is located?

Students: In the right lower lobe.

SQUIRE recognizes that this is incorrect because "Right Middle Lobe" is the correct location for the lesion in this example. However, the program waits until after the students justify their decision to provide feedback.

SQUIRE: What signs do you see which indicate the lesion is located in the Right Lower Lobe?

Students: Because the right heart border is obscured.

At this point SQUIRE recognizes that although the students have correctly identified the obscured right heart border, they have reasoned incorrectly about the location of the lesion (because "Right Heart Border Obscured" does not point to Right Lower Lobe" in the lists in Figure 2). SQUIRE will now tell the students which sign does match "Right Lower Lobe" and then ask again for the location which matches "Right Heart Border Obscured."

SQUIRE: No, if the lesion was located in the Right Lower Lobe, you would probably see the Right Diaphragm Obscured.

But where is the lesion located if Right Heart Border Obscured?

Students: in the right middle lobe.

This is the correct response so SQUIRE can now confirm it.

SQUIRE: Yes, Right Heart Border Obscured indicates the lesion is located in Right Middle Lobe.

We also wanted the dialogue to capture the phase of our model where alternative locations are generated and tested. This is an essential component of diagnostic reasoning in radiology. In order to generate dialogue relating to this process, it was necessary to specify another list

of locations (list #3 in Figure 2) containing negative alternatives. These negative alternative locations are specified in the author program in the same fashion as the correct location. Linked to these alternative locations are the reasons (signs) which enable the student to rule out each alternative (list #4).

The dialogue which SQUIRE can now generate for these alternatives might proceed as follows:

SQUIRE: Do you think there is an alternative location for the lesion?

Students: No.

SQUIRE: Do you think the lesion could also be located in the Right Lower Lobe?

SQUIRE is now asking the students to consider an alternative from list #3.

Students: No.

SQUIRE: Why not? What sign do you see which indicates the lesion is not located in the Right Lower Lobe?

Students: The right diaphragm is intact (not obscured).

The students have given the correct reason for ruling out Right Lower Lobe. SQUIRE now gives feedback and asks about another alternative from list #3.

SQUIRE: Yes, Right Diaphragm intact would indicate the lesion is not located in Right Lower Lobe.

Do you think the lesion could also be located in the Right Pleural Space?

Students: Yes.

SQUIRE: No, there is a sign on this film which indicates that the lesion is not located in the Pleural Space. What sign rules out the Right Pleural Space?

Students: Air bronchogram sign.

SQUIRE: Correct, the Air Bronchogram Sign indicates the lesion is not located in the Right Pleural Space.

So, we see that SQUIRE generates a substantial amount of dialogue from relatively simple lists. However, the list structures alone are not enough. Another

124 essential feature of SQUIRE concerns the way in which the students' responses (especially incorrect responses) to the questions are recognized and then categorized so that the appropriate feedback can be given, or the appropriate next question can be asked. A language processor is used to recognize the students' responses. These responses are then categorized as correct or incorrect depending on whether they match the items in the lists specified by the author program as correct for that film. In the above example, the students' answer to the question "Where is the lesion located?" is checked against the correct response in list #1 in Figure 2, and then categorized as correct or incorrect. Next, SQUIRE asks "What sign do you see which would indicate the lesion is located there?" Again the students' response is checked, this time against list #2 in Figure 2 and categorized as correct or incorrect. There are four possible combinations of correct or incorrect "location" and correct or incorrect "reason (sign)." These four combinations are represented in the 2 x 2 table below.

	Correct Reason	Incorrect Reason
Correct Location	Subroutine 1	2
Incorrect Location	3	4

An examination of the previous portion of dialogue reveals that the students incorrectly reasoned the location but correctly identified the abnormal sign. This combination would be categorized as cell #3 in the 2 x 2 table shown above, i.e., incorrect location but correct reason. SQUIRE then calls up the appropriate subroutine and tells the students which sign they would have seen corresponding to the incorrect location they named, prompting them to reason the correct location from the correct sign which they saw. (See first dialogue.)

There are two modes of interaction which enhance the impact of SQUIRE; the first is human interaction, the other is human-computer interaction. The human interaction occurs among the students as they examine the radiographs in front of them and come to a consensus regarding the interpretation of the abnormal sign. Only one answer can be given back to the computer in response to a question posed by SQUIRE, and the dynamic interchange which occurs in trying to reach this decision stands in contrast to other more isolated and competitive aspects of medical education. The human-computer interaction involves the sequence of questions and feedback which was described earlier. SQUIRE's dialogue was designed to sound as natural as possible, simulating the interaction between doctors and students. We

are currently exploring the possibility of interfacing a speech synthesis mechanism so that SQUIRE can actually "ask" questions of the group and thereby improve the naturalness of the interaction between the students and the computer.

References

- Chase, W. and Simon, H. A. The mind's eye in chess. In W. G. Chase (Ed.), Visual Information Processing. New York: Academic Press, 1973.
- Collins, A. M. Processes in acquiring knowledge. In R. C. Anderson, R. J. Spiro, and W. E. Montague (Eds.). Schooling and the acquisition of knowledge. Hillsdale, N. J.: Laurence Erlbaum Associates, 1977.
- deGroot, A. D. Thought and choice in chess. The Hague: Mouton, 1965.
- Elstein, A. S., Shulman, L. S., and Sprafka, S. A. Medical problem solving. Cambridge: Harvard University Press, 1978.
- Felson, G., Weinstein, A. S., and Spitz, H. B. Principles of chest roentgenology. Philadelphia: W. B. Saunders, Co., 1965.
- Garland, L. H. Studies on the accuracy of diagnostic procedures. American Journal of Roentgenology, 1959, 82, 25-38.
- Gibson, E. J. Principles of perceptual learning and development. New York: Prentice-Hall, 1969.
- Gibson, J. J. The senses considered as perceptual systems. Boston: Houghton Mifflin Company, 1966.
- Newell, A., and Simon, H. Human problem solving. Englewood Cliffs, N. J.: Prentice Hall, Inc., 1972.
- Paul, L., and Juhl, J. The essentials of roentgen interpretation. Hagerstown, Md.: Harper and Row, 1972.
- Squire, L. F. Fundamentals of radiology. Cambridge, Mass.: Harvard University Press, 1975.
- Squire, L. F., Colaiace, W., and Strutynsky, N. The chest. Philadelphia: W. B. Saunders co., 1971.
- Squire, L. F., Blotnick, V., and Becker, J. Self-instruction in radiology for medical students. Radiology, 1972, 105, 681-684.
- Squire, L. F. and Becker, J. On the importance of teaching undergraduate radiology. Radiology, 1975, 117, 227.
- Squire, L. F., Twersky, N., Pals, J., and Becker, J. More effective devices for teaching undergraduate radiology. Radiology, 1975, 117, 63-65.
- Tuddenham, W. J. Visual search, image organization, and reader error in roentgen diagnosis. Studies of the psychophysiology of roentgen image perception. Radiology, 1962, 78, 694-704.
- Supported by a grant from the Medical Education Foundation at St. Paul-Ramsey Hospital and the Consulting Group on Instructional Design, University of Minnesota.

RANDOM GENERATOR FOR ARITHMETIC EXPRESSION PROBLEMS

James D. Jensen

National Technical Institute for the Deaf
Rochester Institute of Technology

A BASIC language program which randomly generates arithmetic expressions is described in this paper. The program is used for creating homework examples and exam questions for a first course in computer programming.

Length and complexity of the expressions as well as frequency of each character within the expression are controlled by parameters within the program. A truth table controls the form of the expression so that each expression is well formed. The numerical value of each example is computed to produce an answer key.

Each example is unique, in contrast to programs which randomly select examples from a data file. This and similar techniques may open the way to more diverse and effective instructional strategies in mathematics, language, and other subjects.

In learning computer programming fundamentals, one of the elementary skills to be mastered is stating an arithmetic expression in the form and notation used by the computer. As in most subjects involving mathematics, drills and evaluations for teaching this skill require an ample supply of problems for the learner to solve. Automating the process of generating examples for this purpose is an attractive idea, not only to avoid the tedium of creating them manually, but for several other reasons.

A major advantage of problems generated randomly by the computer is that each learner can be given a unique, but equivalently difficult, set of problems. Homework and, especially, take-home quizzes to be performed on-line are situations where these equivalent sets of problems are useful. A random example generator may also be useful in some kinds of CAI programs.

Machine created examples differ importantly from humanly generated ones in being truly random. If the example sets are large enough, say fifteen examples per set, the sets are on the average, of equal difficulty and the level of difficulty is consistent over a period of time. Also there is a likelihood of any of the possible combination of elements to be produced. It is difficult or impossible for a mental process to achieve this randomness and comprehensiveness and the result is unfairness to the learner and invalid evaluations of the learning.

The example-generating BASIC language program, EXGEN, described here, randomly generates correctly written arithmetic expressions and prints them out in lists of fifteen together with their numerical values. Others, faced with a similar need for example lists in Math or English, have devised programs which draw

randomly from a file of examples or from a file of elements to be inserted into examples of a particular form, producing a unique set of problems for each occasion. EXGEN, however, creates a set of entirely new examples each time.

The process begins by selecting, according to a number produced by the RND function, one of five categories of elements which can be strung together to form an arithmetic expression. A second random draw selects the particular character or group of characters which will begin the expression. The process repeats, concatenating each newly selected element to the existing alphanumeric string. The length of the string is also determined by chance, but within limits.

To produce a correctly formed expression, the order from left to right in which the elements appear must follow certain syntactical rules. For example an arithmetic operator cannot immediately follow a left parenthesis or another operator, a left parenthesis cannot follow a digit and so forth. The five categories of elements developed on the basis of these rules are: digits, operators, right parentheses, left parentheses, and intrinsic functions.

A truth table (fig. 1) contains the rules for all possible successions of elements and the rules as to which elements can begin or end expressions. The Truth table is represented in the program by a matrix in which the row number is the number of the category to be tested and the column number has been set by the previously drawn element. If the category drawn is rejected by the truth table test, the program branches back for another draw.

The string expression is complete when its length is within the limits determined by the program,

the last element is one which can validly end the expression and right and left parentheses have been matched. Length is controlled randomly between a minimum of seven and a maximum of approximately ten elements. If the ending element is invalid, according to the truth table, building of the string continues. Right parentheses are then added until there are a sufficient number to match left parentheses, according to counter variables in the program.

		1	2	3	4	5	6	7
		S	D	O	L	R	F	E
2	D	1	1	1	1	0	1	1
3	O	0	0	1	0	0	1	0
4	L	1	0	1	0	0	0	0
5	R	0	1	0	0	1	0	1
6	F	1	0	1	1	0	1	0

S = Start R = Right Parenthesis
 D = Digit F = Function
 O = Operator E = End
 L = Left Parenthesis

FIG. 1 - TRUTH TABLE

Numerical evaluation of the expression is useful in grading papers submitted by the learners. Several ways of writing the expression may produce the same result and, therefore, be acceptable. So, if the numerical value of the expression can be produced as part of the output, some labor can be saved.

Evaluation of the expression is not so readily achieved as might at first be expected, however. The expression is in the form of a string variable which cannot itself be mathematically evaluated by the BASIC compiler. The solution is to create a file (EXLIST) into which the main program (EXGEN) writes program lines coded in BASIC language - a program written by a program. Each of these lines contains a PRINT statement and the arithmetic expression written both enclosed in quotes and without quotes, as well as a line number. When the program is run, by a command in the main program, each expression is printed as an alphanumeric string followed by its numerical value. The last statement in EXLIST is a command to run EXGEN, creating another example list.

Another feature of the program is the control of the frequencies with which each of the five categories is chosen. If they were chosen with equal frequency, the proportion of, say, functions to digits would appear unnaturally large. By forcing some categories to be chosen more frequently than others, the resulting expressions appear more like those encountered in real applications. A refinement of this

program would be a frequency distribution tabulated from a sample of applied computations. The program would then model them faithfully.

A number of lesser problems were encountered in getting EXGEN to perform well. Interrupted runs at first resulted from several causes. Values too large for the computer were produced, generally because of large exponents in the expressions. Negative bases to non-integer powers or square roots of negative numbers also brought error statements. These problems were lessened to a tolerable level by a few additional branches in the program.

Translation of the examples from the computer print out to handwritten notation is necessary prior to dissemination to the learners. Although automation of this step would greatly increase convenience and reduce human error, it has not been done because it would require much more elaborate programming and graphic capability. It is felt that the program is justified by the other ways it enhances instruction.

Reduction of labor by using computers in instruction has often been an idea having more promise than fulfillment. The cost effectiveness of generating examples by computer has, like other kinds of instructional computing, many facets. Depending upon the level of use, it may be cost effective in a given situation due to a savings in labor. More likely, its real contribution is in opening the way to more diverse and effective instructional strategies. Techniques such as the one described may produce useful results when applied to Math, Language, or any subject in which a syntactical structure can be found.

Program listing may be obtained from the author at the conference or by writing:

James D. Jensen, Assoc. Prof.
 RIT/NTID
 1 Lomb Memorial Drive
 Rochester, NY 14623

The Significance of Uncontrolled Variables on Research Outcomes of Computer Assisted Instruction

J. F. Huntington
Miami University

Adequate research designs can control many of the effects of extraneous variables through treatment and control group comparisons. Sources of internal invalidity can be managed when researchers assume that there are no differential effects between interactions of sources of internal invalidity and research procedures. Interaction effects which involve external sources of invalidity are more problematic. This paper describes common research practices used for evaluating computer assisted instruction which may result in misinterpretations of results. Interaction, reactive and interference effects have been related to procedures and interpretations. The focus of the paper is toward communicating research design and procedural implications to the CAI practitioner.

The researcher conducting comparisons of the effects of Computer Assisted Instruction (CAI) with traditional instruction is never really certain that the threats to internal validity are adequately controlled. Some types of influences that may threaten the internal validity of research designs were proposed by Campbell and Stanley (1963) to help researchers identify those relevant factors which might confound the effects of the treatment variable. For the purposes of this paper, CAI is considered to be the treatment variable while traditional instruction administered by humans will serve as the control variable.

The assumption that sources of internal invalidity are adequately controlled is based on an additional assumption that group equivalence can be assured through random assignment or matching procedures. Since many CAI research efforts involve one-shot studies, the power of replications to smooth out spurious nonequivalencies that are generated by chance or measurement errors when groups are formed is lost. If group equivalence of all relevant, significant factors is obtained, the researcher really doesn't come to know the true effects of CAI in the one-shot study because the attainment of group equivalence is a result of a set of procedures rather than real equivalence. Even though rules of equivalence through assignments and matching are followed, the possibility of error in judging equivalence is real. This is especially true when random assignment procedures are followed and systematic factors unknown to the researcher operate to a significant degree during

the one-shot study. When matching is used to insure equivalence of relatively small groups the probability of intrusive factors generated by subject attributes which can not be equally partialled out between groups increases.

When control and treatment (CAI) groups represent replications of studies it can be stated with more assurance that threats to internal validity have been adequately controlled. Researchers should be particularly cautious about accepting the assumption that one-shot studies adequately control factors of internal invalidity. Rosenshine's (1974) approach to evaluating research reports about teacher competencies is an appropriate method for evaluating research about the effectiveness of CAI. The approach involves a search for consistencies of findings over a period of time. Essentially, it is a test of the effects of replications of studies investigating very similar factors. This approach treats the effects of history as an external factor to a particular study but one which may impact significantly and differentially from one study to another over time. This occurs when students become more computer-wise and problems arise when the interaction of selection and the treatment differ from one period of time to another and the researcher assumes that the groups are equivalent. Replications permit conclusions about research outcomes over time to be generalized in the sense that applicability is justified to the greatest number of students. If trends have been set, however, the degree of applicability is essentially a function of the fit between the

128 present and the best match of it with the past.

The issue of whether the previous discussion is really a discourse about the differences of the static-group comparison and pretest-posttest control group design focuses upon the interpretation of matching and pretesting. If matching is accomplished without testing, the researcher must assume that the matching procedure has been based on all relevant factors but there is no testing that the groups match on the basis of the measures which are used to determine the effects of CAI. In other words, group equivalence is removed somewhat from the measure of effect. The researcher must rely on assumptions rather than empirical evidence. If pretests are used to check group equivalence or to set a baseline from which comparisons can be made, there is a probability that this testing may have a reactive effect. It is easier to visualize this effect when it is realized that CAI is virtually certain to be categorized as the experimental or treatment variable. It is not possible to pretest without the potential reactive effects of traditional instruction and with a different reactive effect for CAI since the subjects have never experienced CAI. The source of external invalidity appears to be assured through the reactive/interactive effect of pretesting. This effect can be reduced by selecting subjects which have experienced both traditional instruction and CAI, a practice which is relatively uncommon in studies about the effects of CAI.

Subjects without CAI experiences may also be subject to unusual reactions to the application of the treatment variable. It is not unusual for reactive effects to occur with this type of subjects. Generalizations about the effects of CAI are compromised when part of the effect is generated by the experiences of novelty. The interaction of selection and the experimental treatment is unequivocal across groups of different socioeconomic status and geographical location. CAI studies conducted in urban areas among the poor use subjects with significantly different susceptibilities to the treatment than subjects from wealthy suburban areas who have greater access to computers in their schools and homes. Generalizations about outcomes of CAI research should be tempered by the interaction effects between selection and treatment.

Data collected during CAI research are commonly gathered in special places where computer terminals are located. Subjects are moved from classrooms to the CAI room where they experience the treatment. There is little doubt that school subjects recognize that they are being

treated differently. These Hawthorn effects have pervaded CAI research from the beginning. These reactive effects of experimental arrangements are generated by subject relocation, awareness of participation in an experiment and some subjects' delight in leaving a classroom that has become routine and/or unpleasant. These effects may contribute to the measures of CAI performance but not to those of the control. It has not been a common practice in CAI research to also remove the control group from the classroom and provide their traditional instruction while the experimental subjects are receiving CAI. An additional reactive effect is likely when subjects are treated singly or in small groups because few instructional stations are available. Grouping and ungrouping should occur equally in control and experimental groups.

Multiple-treatment interference is assumed to occur when several treatments are applied to subjects. Multiple treatments can take the form of experimenter induced treatments or implicit treatments which occur as a result of differing experimental arrangements. In any event, when multiple treatments occur and some of them are not specifically related to CAI the researcher's ability to tease-out the induced effects of CAI is compromised. When CAI experiences are scheduled over time the sequential exposures may also interact among themselves as well as with non CAI implicit treatments.

This paper has identified sources of external invalidity which are related to research designs and procedures. Adequate internal validity was assumed only after replications of studies permitted smoothing out errors generated by inadequate group equivalencies represented by the one-shot study. The practice of categorizing CAI as a treatment rather than a control undoubtedly leads to generating research invalidities that are derived when an experience is usually viewed as exceptional or unusual rather than commonplace. The application of the treatment (CAI) can also lead to error generation through differential treatment of location, grouping, etc. Until CAI is practiced extensively, researchers should be particularly careful about interpreting comparisons of its effects on learning with traditional instruction.

The following questions should be addressed by the CAI investigator:

1. Will the comparison be conducted as a cross-sectional or longitudinal study?
2. Is group equivalence related in all relevant aspects to treatments and controls. Is CAI always the treatment?

3. Has pretest sensitization occurred?
4. Does the factor, CAI represent a differing curriculum application which could be applied without CAI (i.e. measures of method effects rather than CAI effects)?
5. Have intrusive factors of grouping arrangements, terminal limitations and availability and location-specific factors been adequately controlled?
6. Are the measures of effects relevant and adequately discriminating when comparisons are evaluated?
7. Are subjects learning anything that is meaningful?

Note: Supplementary materials are available from the author.

References

- Campbell, D. T. & Stanley, J. C. Experimental and Quasi-Experimental Designs for Research. Chicago: Rand McNally & Co., 1963.
- Rosenshine, B. Teacher Competency Research. Paper presented at the annual meeting of the Midwest Association of Teachers of Educational Psychology, 1974.

Bibliography

- Cartwright, G. F. Social, Personality, and Attitudinal Dimensions of Individual Learning with Computer-Assisted Group Instruction (Doctoral Dissertation, University of Alberta, 1973).
- Ellis, A. B. The Use and Misuse of Computers in Education. New York: McGraw-Hill Book Company, 1974.
- Hicks, B. Notes on Recent Educational Applications of Computers. The Illinois Series on Educational Applications of Computers, The University of Illinois, #15, 1978.
- Johnson, M. C. Educational Uses of the Computer. Chicago: Rand McNally & Co., 1971.
- Kirk, R. E. Experimental Design: Procedures for the Behavioral Sciences. Belmont, CA: Wadsworth Publishing Co., 1968.

Knights, R. M. & Adler, D. R. A Feasibility Study of a Remote Computer-Assisted Developmental Assessment of Children Via Satellite. Research Bulletin No. 16, Department of Psychology, Carleton University, 1977.

Mason, E. J. & Bramble, W. J. Understanding and Conducting Research/Applications in Education and the Behavioral Sciences. New York: McGraw-Hill, Inc., 1978.

D002 Computer Assisted Instruction. Smithsonian Science Information Exchange, Inc. (a computer bibliographic search). Washington, D. C., 1976.

THE TAMING OF A MINIMICROMAX

Leighton A. Price and Raymond J. Kalush, Jr.

Computer Institute for Social Science Research
Michigan State University

Computer assisted design of mark-sense questionnaires, tests and psychological measures yields high quality instruments, accommodates last minute changes and provides rigorous control over data preparation. This automated approach is useful in connection with both instruction and research. What must be determined, however, are the relative merits of implementing such software on mini, micro or maxi computers. The present paper constitutes a case study of time, labor, convenience, resource and other factors which ultimately determine the cost-effectiveness of its implementation.

A BEAST OF A PROBLEM

In today's university computing environment, faculty wanting to develop applications useful for both instruction and research often find themselves face-to-face with a real beast of a problem. As they gather information and opinions, it becomes apparent that many complex factors must be weighed.

Is a larger computer necessarily better for my application?
Must there be interaction or will batch-processing suffice?
Is a timeshared or a dedicated system required?
What programming language is best-suited to the situation?
Can I afford to use the software once it's completed?
Are there various possible approaches to implementing it?
Will people find the software easy to use?

There was a time when faculty simply had to decide whether it was reasonable to use the university's batch-processing computer. Clearly, the decision process facing people today is far more complex.

As computer technology has advanced and the importance of computers has increased, the beast has grown. Sometimes it has evolved gradually under environmental pressures. At other times, new parts have developed rather suddenly and taken people by surprise.

For lack of a better name, we termed the beast a minimicromax, and the purpose of this paper is to discuss some of our efforts to tame it.

INITIAL ENCOUNTERS

Over the past several years, the Computer Institute for Social Science Research has implemented some interactive software. The majority of this was in connection with the institute's Behavioral Science Instructional Laboratory (BSIL).

Some of these programs were primarily for instructional use (e.g., an experiment simulation program, a system for managing instructional software for courses, problem-solving programs in the social sciences, lessons on psychological statistics). However, in a university environment, the boundaries between instruction and research become difficult to identify. As a result, there was greater interest in interactive programs which could support either instruction or research (e.g., rapid generation of frequency tables from data bases, demographic projection from information supplied by the user).

Initially, BSIL concerned itself with research methods courses in the College of Social Science, and interactive software was implemented in FORTRAN on the university's large Control Data 6500 computer. It soon became apparent, however, that neither our specially designed programs nor the tremendous speed of the CDC 6500 could adequately yield a few simple analysis tables at several terminals simultaneously. Response times for simple interactions were undesirably long (especially toward the end of a term). The cost of student sessions was surprisingly high considering the proportion of time spent thinking. Busy signals frequently made it difficult to handle group presentations. Students were often overwhelmed by the complexities of using a large com-

puter.

As we were experiencing these problems, faculty elsewhere in the university were having similar problems with instructional support software. Fortunately, negotiations with our Computer Laboratory eventually led to the purchase of a 32-port Hewlett-Packard 2000 Access System. The cluster of terminals from Social Science was then hardwired to the system. This made it possible to use both low-speed printing terminals and high-speed video display terminals.

We found the timeshared environment of the HP 2000 computer far more compatible with student needs. It was easier to teach students to use the software, program responses to simple questions were rapid and analysis tables were printed about as soon as they had been by the CDC 6500. We were delighted because it had been difficult to anticipate the impact of shifting from a compiled program environment to one where each line of code is interpreted.

These outcomes encouraged us to develop other social science applications which spanned both instructional and research uses (e.g., automated administration of psychological inventories, simplified text processing at video display terminals, computer-aided questionnaire design).

In the course of developing these applications we encountered the same problems that challenge developers of purely instructional software. We found, however, that many problems have greater impact when a program has to accommodate both instruction and research. Issues can somehow take on a more urgent tone, an operation that works well for a single person becomes a disaster when several people do it simultaneously, etc.

A POINT OF REFERENCE

In our opinion, the Survey Research System (SRS) developed by the Computer Institute constitutes a prime example of how difficult it is to satisfy the potentially conflicting demands of both instructional and research uses. Furthermore, the various modules within SRS place such vastly differing demands on resources that portions of the system resemble several sorts of instructional/research software on the HP 2000.

SRS is an interactive system which supports the design, production and processing of questionnaires, tests, psychological inventories and other instruments. The primary objectives of the system are to: enable professional quality questionnaire cards to be developed

quickly and economically, to eliminate many of the time-consuming and expensive aspects of data preparation, to gain a high degree of control over common sources of error that can plague researchers and to facilitate data analysis.

After rough drafts of questions have been written, a questionnaire design template is used to aid tentative decisions regarding the arrangement of question text and response boxes. Specifications for the questions on each face of a questionnaire card are then put onto the HP 2000 computer via an interactive terminal. SRS asks for information concerning question labels, question types, box locations, box sizes, response codes, etc. Since SRS imposes very few constraints on question organization, questions can be structured for ease of response and the overall layout can be easily revised.

Camera-ready copy of a questionnaire is produced as follows. The SRS user requests generation of plot specification files from card specifications and transfers them electronically to the CDC 6500 computer. This information is then fed into a plotting program to obtain very precise double-size plots of card layouts.

Question text is entered/edited on the HP 2000 via a video display terminal. The question text is then printed on the double-size plot using a high-quality printing terminal.

The double-size plots containing the finished questionnaire are taken to a printer. The plots are photo-reduced, offset printing plates are produced and cards are printed. The offset printing process yields cards which are precise yet reasonably priced.

As questionnaires are returned, bunches of cards are read with an optical mark-reader. SRS uses the specifications prepared in the card design phase to identify cards with errors (e.g., incorrectly stacked cards, single response questions with more than one response, embedded blanks in numeric fields, etc.). Diagnostic messages are printed as errors are identified, and they are corrected interactively.

The prepared data are automatically stored on an HP 2000 file as indicated in the codebook for each face. The user can also enter value labels for questions and automatically generate certain commands needed to analyze the data with the SPSS system available on the CDC 6500. The data file and the file of SPSS commands are then transferred electronically to the CDC 6500 computer for analysis.

It should be evident from this description that SRS is a many faceted system and, as such, places vastly different demands on computer resources depending upon whether it is used for instruction or research, whether one is developing or processing questionnaires, whether question specifications or question text are being entered/updated, whether data are being read or errors are being diagnosed, etc. In addition, the particular mix of applications running on the HP 2000 can interact heavily with all of these factors.

As questionnaire specifications are entered or displayed, for example, the user constantly interacts with the program and relatively little computation or file handling occurs. This is very similar to the way that many simple problem solving or simulation programs operate. As questionnaire specifications are revised, however, array searches and array modification tasks cause the system to become more compute-bound. The operation of SRS then resembles the more sophisticated instructional simulations that require several time-slices to complete calculations.

When question text is entered or updated, a rather different situation prevails. After some simple interactions with the program, there is a moderate amount of file handling followed by a period of time when SRS makes virtually no demands on HP 2000 resources. The terminals we use for this task allow full screens of text to be edited before anything is sent back to the computer. When text is sent, however, mainly file operations occur. These operating conditions are virtually identical to those of our heavily-used text processing software.

The card reading task is quite simple. Card input is requested, a few simple checks are performed and unprocessed data is written to a file. However, in the subsequent task of processing data, varying demands are placed on computing resources. As cards are processed for errors, SRS alternates between requesting input from the user and performing considerable computation as well as frequently reading/writing moderate amounts of disk information. While these operations bear considerable similarity to CAI programs which present lessons and update performance information, they differ in one very important respect. In CAI programs interactions with users predominate. In the card processing portion of SRS, however, interactions with users constitute a much smaller proportion of the activity.

SOME SIGNS OF TROUBLE

Over the past few years, we have had opportunities to observe how timeshared programs are affected by the type of load on the HP 2000. Some of these observations are summarized below.

1. Purely instructional programs (e.g., simulations, problem solvers, etc.) which continually interact with students and make little or no use of files are ideally suited to the timeshared environment of the HP 2000. These applications make brief and varied demands on computer resources, and heavy use at several ports does little to degrade system performance.

2. Programs which access or update considerable file information during an interactive session (e.g., drill and practice systems, our software for producing individual frequency tables, lessons written with the Instructional Dialogue Facility) are also reasonably compatible with the timeshared environment. In a group demonstration, however, certain file operations may be performed by several people simultaneously, and the detrimental effects for these users can be severe.

3. Interactive programs that have compute-bound sections are adversely affected by the number of users on the system and by the extent to which other programs are computer-bound themselves.

4. Interactive programs that frequently reference disk files for small amounts of information do not interfere with one another very much. On the other hand, when programs of the second type mentioned above are used, the detrimental impact on these seemingly simple file references can be quite dramatic.

Unfortunately, several of our instructional/research programs encounter the fourth situation mentioned above (e.g., the automated presentation of psychological inventories, the simplified text processing, automated design and analysis of questionnaires). Under these circumstances, the response times for apparently simple tasks can become rather long. Convenient use of these programs can depend heavily upon the mix of applications running at a particular time, and it is very difficult to anticipate what this will be.

Perhaps the most striking example of the problem occurs in the card processing module of SRS. At times, a simple questionnaire can be processed in a couple seconds on a moderately loaded system. When several programs of the second type mentioned above are in use, however, it can take as much as a minute to process

the same questionnaire. This module accesses small amounts of disk information very frequently and the other programs are simply hogging the disk.

When the mix of HP 2000 applications has severe impact on the card processing phase of SRS, very poor use is made of the user's time. The system runs slowly and data errors are detected only periodically. Although processing can also be done in an "unattended" mode where cases with errors are rejected, the labor costs of locating the cases, correcting errors and reentering them may be greater than the costs of correcting cards interactively.

A POSSIBLE SOLUTION TO OUR PROBLEMS

Although most of our instructional/research applications place relatively light demands on HP 2000 resources, it is obvious that some of the more traditional instructional applications can cause serious problems for them.

With regard to SRS, one possible approach was to implement it as an interactive program on the much faster CDC 6500 computer, but we decided against this for three important reasons. First, our earlier experience with simple interactive analysis of data bases led us to expect unsatisfactory response times and unnecessarily high processing costs for straight-forward tasks. Second, the questionnaire design phase of SRS is most effective when a person can work at a terminal until satisfied with the questionnaire being developed. The cost of connect time and compute time on the CDC 6500 would change this computer-aided process into a high pressure operation that would certainly discourage its use. Third, we thought it very important to have both the design and processing phases of SRS implemented on the same computer. These two segments of SRS are far too interdependent to conveniently implement them on different computers.

For several months, we studied the hardware and software requirements of SRS. We examined many of the human and economic factors pertinent to effective use of the system. In addition, we made careful comparisons of relevant hardware and software capabilities on the CDC 6500, the HP 2000 and a wide variety of micro-computer systems on the market today.

Perhaps the most important outcome of this effort was the discovery of a more generalized structure for SRS. Using this structure, significantly less program code should be required, several major difficulties with the system may be eliminated, it should be easier to

implement the system on various size computers, the system should run considerably faster, less file space should be required, and others.

While some of these considerations caused us to consider reprogramming SRS for the HP 2000, there were many others which ultimately resulted in a decision to implement it on a dedicated micro-computer. For example, the cost of an appropriate micro-computer is about the same as the "per port" cost of the HP 2000. SRS and any of our other instructional/research applications can be readily converted for available versions of BASIC. The version of BASIC best suited to SRS includes some major improvements over HP 2000 BASIC. The processor speed of some micro-computers is sufficient to net an improvement over timeshared use on the HP 2000. Floppy disks can provide adequate on-line data storage for SRS. Information can be conveniently transferred from micro-computers to other computers and vice versa. The portability of a micro-computer expands its utility for both instruction and research.

On the basis of such considerations, we decided it would be desirable to implement SRS on a particular type of micro-computer. The computer purchased for this is a North Star Horizon system with two RS-232 I/O parts (baud rate selectable to 9600), 48K of random access memory and two dual-density floppy disk drives. The operating systems and languages available on the computer are the North Star disk operating system (NDOS), North Star BASIC, the CP/M disk operating system, CBASIC2 (a precompiling version of BASIC) and FORTRAN IV. The North Star NDOS and BASIC are very easy to use, but CP/M and CBASIC2 are essential to successful implementation of SRS. While CBASIC2 is rather different from HP 2000, it is closer to other versions of BASIC and has many important features (integer and floating point arithmetic, sequential and random files, automatic chaining to other programs, formatted I/O, common variables, machine language calls, cross-referencing, etc.).

SOME LESSONS WE HAVE LEARNED

In the course of developing a version of SRS for the North Star micro-computer, we have learned a great deal about the minimicromax. Some of these lessons pertain solely to the micro part of the beast. Others have broader implications for the entire beast.

Training is easier if you feed it serial

The minimicromax has a finicky appe-

tite. The problem is to broaden its tastes without giving it indigestion.

A computer which supports RS232C serial connections with other equipment is exceedingly useful. In particular, hardware and software for serial interfacing provides a convenient means of communicating with terminals, micro-computers, printers, larger computers, modems to other computers, card readers, etc. In general, there is little fuss or bother involved in making the equipment work.

Micro-computers which also have a serial console port have many advantages. Any type of computer terminal (printing or video) with a serial connector can be used as the console. The system is, therefore, highly versatile and few applications fall beyond ones reach.

Don't be fooled by a few pleasant phrases

It's deceptively easy to get the beast to say a few simple phrases. If its pushed though, it may simply grunt and stutter.

In the case of our micro-computer, we discovered that the serial I/O routines supplied with the system operated well with 110 baud printing terminals and all types of video display terminals. They did not, however, work properly with many older printing terminals running at 150 baud, 300 baud, etc. (e.g., TI Silent 733). Characters at beginnings of lines were lost because the computer sent information before the carriage had time to return. The only printing terminals which worked properly were ones that buffer lines of input.

We also found the standard I/O routines deficient when used with terminals having block or format mode. When a full line was transmitted to the micro-computer, the character input and echoing processes stepped on one another. By modifying the I/O routines so the character echo could be turned off manually, the problem was overcome.

When a mark-sense card reader was used with the micro-computer, the character echoing problem again caused difficulty. In this instance, however, it was necessary to turn echo off before each card was read and turn it on immediately after reading the card so diagnostic messages could be output to the terminal. Although the BASIC lacks an echo on/off command, some careful POKEing of the I/O routines yielded the necessary program control of character echoing.

Know its blood type or suffer the consequences

It is extremely important to know what will flow through the beast's system. As with human blood types, there are universal donors and universal recipients but there are also situations where a mistake can prove very costly.

It is popular to speak of downloading and uploading. While this frequently refers to moving programs from one computer to another, it also encompasses the transfer of data and text. Discussions frequently carry the implication that it is all very easy to do. We should like to caution otherwise.

Given serial interfaces, the ability to receive/send ASCII information and the ability to store what is received/sent, it is relatively simple to transfer data or text from one computer to another.

When it comes to transferring BASIC programs, however, problems can start to show their ugly heads. Under many operating systems, source programs must be entered at the console port and debugged as they are entered a line at a time. Furthermore, programs are stored on such systems in compressed code (P code) that may differ from one company to another. Obviously, this creates problems if you have source programs on a larger computer that you want to run on a micro-computer. The fact that the small computer may output programs to a serial port in source form is small consolation when you want to go the other way.

Watch out for slipped disks

It is extremely difficult to train the beast to do fancy tricks when it finds the simple matter of walking painful.

A dual disk micro-computer is not a luxury, it is nearly a necessity. While it is possible to function for some time with a single disk system, it becomes apparent that dual drives are needed: a) when large programs are being developed, b) for convenient backup procedures c) to provide open-ended file storage, or d) when some of the programs on one diskette need to be added to those already on another diskette. In a single drive system, these and other tasks can be awkward or even impossible.

We have also found that a system such as SRS requires either 8" floppy disks or dual-density 5" mini-floppy disks. There is simply too much source code and data storage requirements are too great for single-density mini-floppy disks to

suffice.

Just because it runs fast doesn't necessarily mean it wins

A sprinter can capture a lot of attention, but a long-distance runner is likely to win our kind of race.

As we have worked on the micro-computer version of SRS, we have taken opportunities to compare the speed of the North Star, the HP 2000 and the CDC 6500. Thus far, comparisons of some compute-bound operations have been made in BASIC. The results are very interesting.

A particular computational task took 18 minutes in HP 2000 interpretive BASIC (7 digit accuracy) under light to moderate load. In North Star BASIC the task took 22 minutes (8 digit accuracy), in "precompiling" CBASIC2 on the North Star the computation took 60 minutes with 14 digit accuracy but only 8 minutes with 5 digit accuracy. Few operations in SRS require more than 5 digit accuracy, and integer mode in CBASIC2 rendered the North Star micro-computer a clear winner over the HP 2000. Although processor speed is much greater on the HP 2000, delays caused by timesharing and overhead put the system at a decided disadvantage.

When the same computations were run with the true compiling BASIC of the CDC 6500, they took only 35 seconds. The reader may remember from earlier discussions, however, that heavily interactive applications were more cost-effective and convenient on the HP 2000 than on the CDC 6500. Clearly, compute-bound timing tests tell only a very small part of the story.

Reading and writing are taught with difficulty

Just because it can use words doesn't mean it can communicate well.

The editing software (for text, data or program code) available for most computers is justifiably the subject of much criticism. Generally, the software is oriented to line-at-a-time entry and updating. The fact that many of today's video display terminals are more sophisticated than the teletype has yet to have much impact on available editing software.

Dedicated word processors have evolved a great deal over the past few years, and they are easy to use for simple applications. Much of this progress is evident in the word processing software available

for general purpose micro-computers.

135

Just such a text editor is an integral part of the Survey Research System. It allows the text of questions to be added or deleted anywhere on the screen of a video terminal having block or format mode. Anyone can learn to use this software in just a few minutes, and it allows people to make very efficient use of their time.

CONCLUSIONS

Thus far we are surviving our encounters with the minimicromax. Sometimes we have found the micro portion of the beast a real challenge. Fortunately, most of our problems have stemmed from unfamiliarity rather than inherent deficiencies.

The North Star micro-computer is turning out to be an excellent choice. Comparisons we made prior to purchasing the computer are proving worthwhile, and our expectations regarding the system are being satisfied.

To date, nearly half the micro-computer version of SRS has been written, and CBASIC2 has been able to accommodate everything required. From the amount of source code and precompiled code stored on disk thus far, it also appears that the entire system will occupy fewer than 150K bytes of storage (100K for source and 50K for precompiled code). This will constitute a significant improvement over the 225K bytes used on the HP 2000 mini-computer.

We must also emphasize that we do not view the North Star micro-computer as a "stand-alone" system. While it is clearly able to handle much of what we need to do, it must be viewed as an integral part of a larger network consisting of mini and maxi computers. Without the ability to communicate with other systems, the utility of a small system must eventually be questioned.

In summary, we have examined some of the problems of taming a minimicromax. We have advanced considerations which seem important to the instructional/research setting in which we are operating. The reader must remember, however, that the beast can look quite different from the perspective of other applications in different environments. Hopefully, the case study presented here will make it easier for others to handle the beast that may sometimes confront them.

The T.I.E.S. Instructional Management Support System

Donald Holznagel and James Sydow
TIES, St. Paul, MN 55113

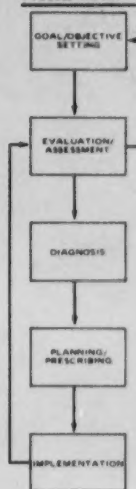
The Instructional Management Support System (IMS) is a computer system providing support for a wide range of instructional management plans. This paper defines the model governing the system, describes the features now completed and discusses components remaining to be implemented. The system is used in elementary, secondary and vocational schools, with information being provided to students, teachers, and school or district level administrators. Student progress according to objectives is monitored, evaluated and reported by testing, observation or both. The system is independent of subject matter, and is now used in many subjects at all three levels mentioned above. The system is important for its generality, which is particularly important for processing centers serving a region with a variety of school types and districts.

Presentations of Instructional Management Support System (IMS) have been previously made by TIES staff members at conventions of AEDS and affiliated organizations (NAUCAL and MAEDS). The purpose of this presentation is to provide updated information on IMS and discuss modes of usage by TIES member districts. The written material included in this paper, for the most part is excerpts from the 296 page User Manual which was released in 1978. This information will serve as the basis for discussing IMS.

An Instructional Process Model

Instruction is a complex process which includes a number of components. These components require a variety of resources and activities - from simple record keeping to sophisticated teaching. The model illustrated and described below is an attempt to classify the components of the instructional process according to their functions, and to describe the activities associated with each component. While the components and activities have been divided into distinct and separate parts for the sake of the model, the actual instructional process usually involves a blending of activities. It is important to keep in mind that the process described is only one model of instruction, and that actual learning situations may deviate in part from this model. The purpose of the model is to provide a basis for analyzing the teaching/learning situation. Thus an instructional management system utilizing computer support may be incorporated into the process where applicable.

An Instructional
Process Model



In the model, the instructional process is divided into five major components:

1. Goal/Objective Setting
2. Evaluation/Assessment
3. Diagnosis
4. Planning/Prescribing
5. Implementation.

Each of these components has a distinct function within the overall process and requires a specific set of activities and resources. The first four components are designed to help direct and manage the fifth component. These four comprise the scope of instructional management.

A number of computer support functions can be provided for the management aspects of instruction. A number of these support functions are included in the description of each component

Goal/Objective Setting

This first step of instructional management is to define overall course goals and specify student performance expectations. A scheme for identifying and organizing the objectives is developed, and the criteria for evaluating student performance in terms of the objectives is established. Computer support functions for this area include analysis of survey responses, storage and retrieval of objectives.

Evaluation/Assessment

Upon completing the initial step, it becomes necessary in the instructional process to determine how the student performs in relation to the goals and objectives, and to assess the instructional procedures and tools used. This may lead to a redefinition or restatement of goals, objectives, or test items or may result in a change in teaching techniques.

The computer's role in this component includes test item storage and retrieval, test generation, test scoring and item analysis, and the generation of status reports, progress reports, data summaries.

Diagnosis

Diagnosis involves the development of a student profile, areas of student strength and weakness, and other variables which may affect the learning process. This component aids the individual teacher, principal or school district in identifying the instructional needs of the students. Diagnosis may include numerous activities, such as a correlation of student ratings in a skill level, comparison of student performance levels with norms, analysis of historical patterns, and correlation with medical or other relevant data. Computer support functions here include the generation of comparison, exception and historical reports, as well as many of the functions described for the Evaluation/Assessment component.

Planning/Prescribing

In determining the educational experiences and settings necessary to meet the instructional needs, this step takes into account student diagnostic information as well as the resources, strategies and activities that are available. This process may involve grouping students, or matching individuals or groups of students with facilities and staff. It also involves the selection of appropriate instructional resources. Computer support includes grouping and scheduling routines for the classroom, school, or district, generation of teacher characteristics profile and instructional material reference storage and retrieval.

Implementation

Finally, a variety of techniques, strategies, and environments may be used in the instruction of the student. Selection is determined by the information gathered in the previous stages. Once instruction is completed, the process again moves through the cycle, beginning with the Evaluation/Assessment component.

Computer Support Functions of IMS

The chart below lists the specifics of the computer support functions incorporated into the TIES Instructional Management Support system. It becomes apparent that not all functions capable of computer support and an integral part of the instructional model are a part of IMS. Some are supported by other TIES systems, while others are in various stages of design and development.

	IMS Supported	Other TIES System	Not Supported
<u>Goal/Objective Setting</u>			
Survey Analysis		x	
Objective Storage			x
Objective Retrieval			x
<u>Evaluation/Assessment</u>			
Test Item Storage			x
Test Item Retrieval			x
Test Construction		x	
Test Scoring	x	x	
Item Analysis	x	x	
Student Progress Reports	x	x	
Student Status Reports	x		
Data Summaries	x		
<u>Diagnosis</u>			
Group Analysis	x		
Student Comparison	x		
Student History Profile (Functions listed in Evaluation/Assessment)	x		
<u>Planning/Prescribing</u>			
Grouping	x		
Scheduling	x	x	
Teacher Profiles			x
Resource Materials Storage			x
Resource Materials Retrieval			x

IMS PROCESSING CAPABILITIES

Before IMS may be used a series of activities must be initiated which build the course data base. Since IMS uses other files in the TIES integrated data base (i.e., Student, Activity, and Personnel), establishing a course in part involves defining the relationship of IMS data to other files.

Having completed necessary preparatory activities IMS is used for two major functions:

Testing: The testing function of the IMS system allows the user to input student response information for a test. After the necessary information is input, the IMS system will score the test. Based on the student response information, six reports may be generated. These reports give a history of individual student testing information, group testing information and test item information. The testing function allows the user to have more than one school within each course to accommodate district-wide testing.

Rating:

The rating function of the IMS system allows the user to input student rating information. The rating information is

138 generally based on an observation of a particular competency or skill. The competency or skill may relate to the cognitive, affective, or psychomotor domains. The rating also may be automatically assigned based on a percentage score of an IMS test. After student ratings have been entered student groups may be formed based on the ratings and four different reports may be generated. These reports give individual student rating information which may reflect the present status or a historical summary, a listing of all rating scale forms and the number of students at each rating, a matrix listing students within a course and their ratings and a student transcript. The rating function also allows the user to have more than one school within each course to accommodate district-wide rating.

It is possible to combine the testing and rating functions through the use of the test linking capability.

Data is input to IMS via several media. Either mark sense or keypunched cards are transmitted from the district via the TIES telecommunication network through the use of a Bell & Howell Model 80 Intelligent Mark Reader (IMR). The Conrac 480/25 Cathode Ray Tube Terminal (CRT) is also used for entering or updating data on the IMS files. This is accomplished through the use of CRT Formats (forms) which provide user capabilities for accessing and updating specific data fields or for initiating processing requests.

The on-line network is available daily to the member districts from 8:00 AM to 5:00 PM. Each evening IMS processing includes the components of editing, updating, and reporting in a batch production mode. The following chart summarizes the reports which are standard products of IMS.

IMS REPORTS

REPORT NO. DESCRIPTION

001 Student Response A record of objectives tested, those answered correctly, and those on which instruction was received Purpose: To provide students with an individual report showing <u>test results</u>	003 Group Summary A record of how well each Student Summary group has done on each tested content group Purpose: To provide teachers and administrators with historical summary of pre-test, post-test, and retention instruction completion information to aid in instructional decision making
002 Teacher Summary A record of each student's performance on all IMS tests taken to date Purpose: To provide the teacher with a complete summary of student response information	

004 Test Form Evaluation A record by question number, of the related objective, the number of responses made, the percent correct, the correct answer, and the percentage of responses to each distractor Purpose: To provide teachers with information to assist them in evaluating and revising test forms	008 Rating Scale Forms and Counts A record of all rating scale forms and their ratings, a count of all students rated on each rating, and all status groups and a count of all students assigned to each status group Purpose: To provide a means of verifying all entered rating information
005 Student Test Analysis A record for each student of the number of responses made and the percentage correct for each objective, unit, or strand Purpose: To provide historical information about student progress	Rating Scale Forms Index The last page of the report which alphabetically lists each rating scale form and status group for the course 009 Group Matrix An alphabetical record of the members of a designated group and their rating on the designated range of rating scale forms
006 Item Analysis A record of the percentage of correct responses to each of the distractors for each test item, categorized as either pre-, post-, or retention Purpose: To provide teachers with the information needed to evaluate and revise individual test items	Purpose: To provide an overview of the ratings for a particular group of students 010 Transcript A record for each student of every assigned rating and the corresponding rating scale form
007 Course Analysis A comprehensive description of all course, student, and testing information stored for an IMS course Purpose: To provide a means of verifying the information entered to date for the testing function of an IMS course	Purpose: To provide a take-home report for students and parents as well as a school transcript 011 Student Status List A comprehensive record for each student of every assigned rating and the corresponding rating scale form Purpose: To provide a means of monitoring student progress

IMS USER MANUAL Instructional Management Support System

The following outline describes a general approach school districts may follow in determining whether or not to use the Instructional Management Support System. It is not meant to detail each step in this process but merely to provide district staff with enough information to clearly understand the type of commitment and the nature and variety of activities necessary to effectively use IMS.

The basic purpose of this outline is to enable district staff to make a reasonable judgment as to the value IMS will have and whether the resources - both dollars and staff - are available to meet IMS course development requirements.

- I. Attend an overview IMS session - conducted by TIES Instructional Staff
 - A. examine the philosophy of the IMS system design;
 - B. discuss the instructional model from which IMS was developed;

- C. examine the IMS structure in terms of its present and long range capabilities, and
- D. discuss the commitment of staff and equipment costs.

II. Review the information obtained in the IMS overview through discussions with:

- A. curriculum directors,
- B. administrative staff,
- C. instructional staff,
- D. others, such as school board members or parents, and
- E. TIES Instructional Services staff.

III. As a result of these discussions, attempt to determine whether:

- A. IMS will assist a district in meeting instructional goals and objectives more effectively and efficiently than other means, or will supplement on-going management to the point where benefits outweigh the non-benefits;
- B. an administrative commitment in terms of financial support and staff development time is present; and
- C. an instructional staff committed to systematic instructional management procedures.

If the result of III is ambiguous or negative, possibly another route toward meeting goals of an instructional program should be explored by the district.

If the response is clear to pursue IMS further, that decision should be conveyed to TIES Instructional Services. Next, the EIS Coordinator and any other district personnel - both administrative and instructional - who will be responsible for IMS course development, should request to attend the IMS implementation workshops.

IV. There are three IMS implementation workshops which outline in a step-by-step process, the methods for organizing and inputting:

- A. course content for the testing function of IMS,
- B. course content for the rating function of IMS, and
- C. student information used for both the testing and the rating functions.

V. To complete the implementation process a TIES Instructional Services staff person conducts a series of in-district sessions for staff who

will initiate the IMS course development.

139

These sessions thoroughly review the IMS manual and give complete training in such areas as:

- A. development of objectives,
- B. development of test items, creating test forms and rating scale forms,
- C. course content structure,
- D. student grouping,
- E. types of reports, their interpretation and use,
- F. methods of recording progress,
- G. the input process,
- H. extract, edit, and error correction functions,
- I. special processing problems,
- J. end-of-year processing,
- K. turnaround cycle, and
- L. training of CRT and IMR operators.

The above areas do not include all the activities which make up an IMS system. Further, due to continual development within the IMS system, various versions are piloted which later become a part of the standard system.

It is essential that district staff responsible for the IMS courses become aware of the potential power available through IMS by attending IMS workshops held throughout the year. In addition, it is necessary that each district's IMS Coordinator, or staff person responsible for IMS development and processing, have a thorough knowledge and understanding of the IMS Manual.

THE TRS-80 AS A CLASSROOM TEACHING
TOOL: TEACHING HEBREW AND RELIGIOUS
STUDIES

Samuel W. Spero
Institute for Computers in
Jewish Life
7135 North Carpenter Rd.
Skokie, Illinois 60077

Two instructional strategies incorporating the Radio Shack TRS-80 microprocessor-based computer system are described. In the first of these strategies the TRS-80 is used to drive a 24" television monitor which is used in the classroom as a "dynamic blackboard." In the second strategy the small, portable, high-speed printer available with the TRS-80 is used to generate printed materials to be used with the entire class. Examples of how these strategies are used in the study of the Hebrew language, Bible, and Jewish laws and customs will be described, as well as the relevance of these strategies for many courses in general studies presently taught in elementary and secondary schools.

In this paper two instructional strategies incorporating the Radio Shack TRS-80 microprocessor based computer system will be described. These same strategies have been described elsewhere (T.H. E. Journal, VI, #2, Feb., 1979) as used in the context of teaching computer programming and mathematics. In this paper the strategies will be applied to instruction in the Hebrew language, Bible, and Jewish laws and customs. These strategies take particular advantage of hardware features available with virtually all of the low-priced microprocessor systems presently on the market.

In the first of these strategies the TRS-80 is used to drive a 24" television monitor. Because of the graphics capability of the TRS-80, as well as its 32 character-per-line output on the television screen, it is possible to have the output of the computer readily visible to all of the students in the classroom. The graphics capability makes it possible to even present words in Hebrew. Using this feature we have been able to develop computer based instructional units in which the computer - with the television monitor - functions as a "dynamic blackboard." Briefly, the computer is used to generate material on the television screen which provides a framework for classroom discussion. Used in this way the conventional classroom with 30 or 40 students has become far more exciting for the students and has led, in the opinion of the teachers using this strategy, to improved learning.

In the second of the instructional strategies to be described, the small, portable, high-speed printer (the QUIKPRINTER which is a Centronics P-1) available with the TRS-80 is used to generate printed materials to be used with the entire class. The materials are used for motivation and enrichment, for reinforcement as homework assignments, and even for evaluation purposes. The materials generated by the computer can be duplicated by the computer itself which prints at 180 characters per second or can be conveniently duplicated by creating a spirit

master from the computer output or making xerox copies. In a number of applications we have even developed individualized sets of materials by having the computer print the student's name in conjunction with his or her assignment. These applications will be described. The use of this strategy has led to increased student motivation and, we hope, increased learning without substantially increasing the work of the teacher.

At the "bottom line" of both of these instructional strategies which incorporate the use of the microprocessor-based computer system is their cost-effectiveness. Compared to all of the other alternative hardware approaches, microprocessor-based systems are the least expensive. Furthermore they permit teachers to do considerably more in terms of various instructional strategies. We are now able to do things in the classroom which we were not able to do before - although we would have wanted to - and even to do those things which we were doing before, but more effectively.

Hardware

Before describing the instructional strategies in detail, the hardware configuration shall be presented. As mentioned, the features available in the Radio Shack TRS-80 are available for both the Apple II and the Commodore PET. What is important to note are the particular features which make possible the instructional strategies to be described. These features have not been conveniently available on conventional time-sharing systems. In presenting these strategies to other teachers in recent months we have found them to be far more receptive than to the strategies implicit in the use of time-shared terminals. (Is there a message here?)

The hardware configuration includes the following items and features:

- a) a CPU including keyboard with 16K bytes of internal storage. We have been able to recommend

- as little as 4K bytes to schools, since many of the computer programs we use to implement the strategies to be described do not require more than 4K.
- b) convenient output to a television monitor. The Radio Shack TRS-80 uses a conventional DIN connector which is readily available at any electrical supply store. The Apple II also has a convenient connection. The PET's connection is a little more complicated. In addition the size of the output on the monitor must be readable some distance from the monitor. We have found that a 32 character-per-line output, which is available on all three of the above systems, is quite adequate. The Digilog Telecomputer is the only time-shared terminal that we know of that has a similar output capability. Note: the Digilog costs in the neighborhood of \$2000. This is more than the cost of the entire Radio Shack system.
 - c) convenient medium of program storage. All of the home computer systems use conventional audio cassette technology for mass storage. This makes it very convenient to transfer programs between schools and between teachers. It is also psychologically satisfying to the neophyte computer person to have complete control over their programs. Also available are minifloppy disk systems which can be used with these personal systems. While the use of this technology adds \$400 to \$600 to the cost of the system, the minifloppy disks have the advantage of greater speed and reliability while still retaining the use of a relatively inexpensive medium for program storage. (A minifloppy disk capable of storing 80 to 120 K bytes costs between \$3 and \$5.)
 - d) high speed printer. All of the home computer systems mentioned support printers such as the Centronics P-1. This printer, for example, prints at 180 characters per second, prints both upper and lower case, and prints in several character sizes. It is quiet, too. Most important it only adds about \$600 to the cost of the total system. Compare this to the popular Decwriter which only prints at 30 characters per second, does not have various character sizes, and costs itself in the neighborhood of \$2000.
 - e) portability. All of the systems mentioned can be placed in their entirety on a cart and be moved from classroom to classroom. In order to operate the system all that is required is an electrical outlet. Compare this to time-shared terminals most of which can only be moved with difficulty and all of which must be connected to a telephone. Another expense.
 - f) ease of service. Any Radio Shack store will accept a TRS-80 for service. They in turn send it on to a local service center - their responsibility, not yours. PET advertises that their system can be serviced by any TV technician. Such is not the case with computer terminals, and certainly not the case with the computers themselves.
 - g) finally, we have alluded to the low cost. A 16K CPU, plus monitor, cassette recorder, expander box, and printer is about \$1900. The addition of the minifloppy disk plus memory brings the total cost to \$2500. The Apple II and the PET are competitive for the same features. **THIS IS THE COST FOR THE COMPLETE SYSTEM!**

Strategy 1 - The "Dynamic Blackboard"

To understand the way in which the computer is used in this strategy, we will first discuss the teacher's use of the blackboard.

When a teacher uses the blackboard, it is generally to create a framework for classroom discussion - even if the "discussion" is primarily the teacher lecturing. Seeing the various salient points of the discussion in print helps the students understand.

Because setting up material on the blackboard can be a time-consuming and tedious process for the classroom teacher, alternatives to the blackboard have been developed. The most popular of these is the overhead projector. Its advantage over the blackboard is that materials for classroom discussion can be prepared ahead of time either professionally or by the teacher himself.

In many ways our use of the computer and the television monitor can be compared to the use of the overhead projector in that the materials are prepared ahead of time for display in class. Also materials can either be prepared professionally or by the teacher himself. The computer does have several advantages. One advantage is convenience. When many overheads are to be used, there is much flipping and sorting. "Flipping and sorting" with the computer are done at the touch of a key. The computer programs are stored on small, audio-cassettes which take up far less room and are easier to carry than the bulky overhead transparencies. Furthermore the presentation of the computer is dynamic. There can be animation; there can be many alternative choices for presentation; there can be teacher-designed graphics; there can be unique and different situations presented each time the "dynamic blackboard" is used.

Interestingly, because we live in a generation of students raised on television, the use of this medium seems to have an unexplainable hold on the attention of the students. Students who would otherwise ignore the blackboard and even an overhead projection will play close attention to the television monitor. This is a phenomenon which we have observed again and again using this approach at many different levels of instruction and in many different subject areas.

Our use of this strategy in teaching mathematics and computer programming have been described elsewhere. In this presentation we would like to describe our application to the study of both Hebrew language and to the study of various aspects of Bible and Jewish laws and customs. The implications of these applications for general language study and for the study of all humanities are fairly obvious.

The applications to be described here have been developed by the staff of the Institute for Computers in Jewish Life, a division of the Hebrew Theological College of Skokie, Illinois. One of the purposes of the Institute is "to explore, develop, and disseminate applications of computer technology which can help improve ... the effectiveness of the services rendered by Jewish Institutions." (from the brochure of the Institute for Computers in Jewish Life)

142 The Radio Shack TRS-80 has been selected by the Institute as the system to be used in the member schools of its Educational Network which includes Jewish parochial schools from many parts of the United States and Canada. The strategies described here have been tested and are being used in the schools of the Network. (For more information contact Rabbi Gerald Porath at the Institute for Computers in Jewish Life.)

Hebrew Language - In the study of the Hebrew language, as in the study of all languages, there are many skills to be taught. These include skills in grammar, spelling, and word recognition. In the study of Hebrew besides the above the students learn a new alphabet. Crucial to the acquisition of language skills is drill-and-practice. The more the students review the materials, the more likely they are to learn them.

In Hebrew grammar all conjugations arise out of a basic root form to which are added various prefixes and suffixes which establish the person and the tense. It is important to emphasize these prefixes and suffixes. One of the "dynamic blackboard" exercises which has been developed to assist in achieving this objective is one in which the computer (using its graphics capability) generates the basic root form in Hebrew and then adds the relevant prefix and/or suffix yielding a particular conjugation. The computer then asks the student to translate. When the correct answer has been given, the computer circles (using graphics) the prefix and/or suffix thereby reinforcing the correct answer for the other students in the class. This exercise has been used in class as a "fun" review, as part of a classroom game in which teams compete, and even as an individual and small group tutorial. In every case we are using the computer as a means to motivate more drill-and-practice. Because of the large size of the characters presented by the computer on the television monitor, there is this flexibility to use the computer in the classroom in a variety of ways.

A "typewriter" program has also been developed for the teacher to use in teaching spelling and word recognition. With this program the teacher can type into the computer any word with appropriate letters being replaced by dashes. The computer will print out this word in very large script. This typewriter program is available in both Hebrew and English. In addition to simply printing out the words with the computer, we have embedded the "typewriter" into several types of games to help motivate the students. In one of these games the class is divided into several teams. The teacher asks each of the teams to spell a word, to determine missing characters, to translate a word, to determine the appropriate conjugation, etc. As each team answers, the computer keeps score. After several rounds the computer prints out a graph of how each of the teams is doing. A second very popular game is "hangman" or an alternative version called "crush-man" where the class has to get the word or words required before "sentence" has been carried out. The graphics make a great deal of difference with the students really being motivated.

Bible Study and Jewish Laws and Customs - A somewhat similar approach has been used in the case of Bible study and in the study of Jewish Laws and customs. The difference is that instead of using only single words, banks of questions with various types of solution are used on the television screen. The "typewriter" program represents the basic framework, in this application, however, entire questions - some with multiple choice answers - are typed onto the screen of the monitor by the computer. In order not to use classroom time, the questions have been prepared ahead of time and stored on cassette. The teacher has merely to type in the number of the question he is interested in using and it will appear on the screen. Furthermore once the teacher has decided on the question, he can also select from various computer responses, choosing whatever he feels to be most appropriate. These responses might be references, the correct answer, a clever reinforcement, etc. In this application we have also used various gaming schemes to improve classroom motivation. The advantage of all of this is that the computer manages the game, motivates the students, and really helps organize the classroom for discussion without adding appreciably to the time required by the teacher for preparation.

Strategy 2 - The Computerized Materials Generator

The average classroom teacher spends considerable time preparing materials of one sort or another for distribution to his or her class. These materials might be extra homework assignments dittoed for the class; the materials might be a quiz or test; they might be extra credit or enrichment materials. The teacher sometimes also prepares "fun"-type materials such as WORDFIND and CROSSWORD puzzles for use by the class to reinforce the various topics discussed in class. To be able to distribute these materials to his or her students, the teacher must first prepare the materials, transfer them to some appropriate medium for duplication such as a ditto-master, and then actually duplicate them. We have found that the TRS-80 can be used for all of the above.

Bible Study and Jewish Laws and Customs - For lack of a Hebrew printer (we are working on this) we have not been able to use the printer as we would have like for the study of language. In Bible study and in the study of Jewish laws and customs where there is not as great a need to work in the Hebrew language itself and where it is possible to transliterate, we have been able to accomplish a great deal with the computer and the attached high-speed printer.

We have prepared question banks and a computer program which generates from these question banks quizzes and tests of various lengths and in as many different versions as a teacher requests. This program can be used for matching, fill-ins, as well as multiple-choice type quizzes and tests. In its use in class this approach of preparing multiple versions of a test has reduced cheating considerably. A little "wrinkle" that we have added to the program types up the test and puts the student's name on it. The program then types up all of the various answer keys indicating which students received which test. Using a code number printed on each test it is very easy to sort the tests (or quizzes)

once they have been completed and then to score each group using the key prepared by the computer. We have recently learned that a company on the West Coast - Chatsworth Data Corporation - is preparing a pencil-marked-card reader for use with the TRS-80 which will make it possible to score the tests automatically as well.

In addition to generating tests we have used the computer to prepare WORDFIND puzzles and other vocabulary games popular with the students. Since in virtually every subject there is considerable vocabulary for students to learn, this approach has proven to be very useful to both teachers of religious subjects as well as the teachers of social studies and science in the various schools in the Institute's Educational Network.

While the use of the computer in this fashion is not particularly new and has been described and documented elsewhere, what is unique here is the convenience and accessibility of the materials generator to the classroom teacher. We estimate that it costs around a penny a page of materials generated by the computer (that includes the cost of the expensive paper used by the high-speed printer attached to the TRS-80). This is competitive with just about any other approach to materials preparation. The relatively high speed of the Centronics P-1 (Radio Shack's Quikprinter) is a major advantage since materials for an entire class can be prepared in a reasonably short time. For those teachers who prefer to ditto or xerox the computer output, we have found that the special paper used with the Centronics P-1 duplicates and xeroxes very well.

Summary and Conclusions

The two instructional strategies we have presented in this paper place the computer into an instructional role for which it is unquestionably well-suited. In this role it provides the classroom teacher with technological assistance in completing tasks which he or she could not reasonably perform himself. To provide this assistance we require only a minimal computer facility - a personal computer, if you will - which is very cost effective. The teacher in using the computer, either as a dynamic blackboard or as a materials generator, reaps maximum benefit. We find that used in this way the computer helps motivate the student; helps the student learn concepts more effectively; helps students reinforce learned concepts more efficiently; and even assists in the evaluation of the entire process of instruction. Using the term in its broadest sense, what more can we ask of computer-assisted-instruction?

It should be noted that neither of the above mentioned instructional strategies involves implicit student use of the facility. It is the teacher who controls the usage of the computer, and it is to the teacher that the computer is dedicated. This represents a major departure from the "traditional" role assigned to the computer in instruction. In presentations to educators from many different backgrounds, we have found a far more positive reception for the potential of the computer when it is used in the context of the above

instructional strategies. It must be made clear to educators that the computer is a tool of instruction for the teacher to use and not just the student. Student use of the computer is not precluded in this approach. Rather student use of the computer is simply one of the several possible alternative strategies which may be selected by the teacher under appropriate circumstances.

Large scale integration of microprocessor-based systems such as the Radio Shack TRS-80 together with reemphasis of the teacher's role in computer usage in schools may well be the key to the long awaited acceptance of the computer as a primary tool of instruction in schools.

USE OF INFORMATION RETRIEVAL PROGRAMS FROM A (IDS TO JOB HUNTING) TO Z (EBRA KEYS).

Janice M. Glime, Ph.D.

Department of Biological Sciences
Michigan Technological University
Houghton, MI 49931

At Michigan Technological University, two information retrieval programs are being used for educational purposes.

The USDA search program FAMULUS is being used for computerized job matching. Students in search of a biology job can list their job interests and background and the program will provide them with a list of potential employers.

The program KEYOUT, written at MTU, sorts taxonomic data to aid identification. It has options which provide for testing, keying, reviewing, checking a data matrix, and updating keys. It is applicable to any taxonomic field and is particularly useful for testing powers of observation.

Introduction

Information retrieval systems are widely used for sorting information such as bibliographies and research data. This paper will discuss less traditional uses for them as aids to undergraduate education.

Aids to Job Hunting

The first out-of-the-ordinary use to be discussed arose when a problem needed to be solved. Our undergraduate biology majors were concerned about finding jobs. The biology faculty decided that a computerized matching system might be appropriate. Since the Harvard program FAMULUS was already available to support the Lake Superior Basin Bibliography, it was selected for the information retrieval.

The data base was designed for use by undergraduate and graduate students seeking summer or full time employment in a biological field. However, its use soon grew to that of providing a counseling tool as well.

To create the data base, one faculty member, working with an undergraduate work study student, designed and mailed questionnaires to potential employers of biologists. The initial list was created by looking through ads in biological journals such as BioScience and Science. Our alumni were all sent questionnaires asking where they were employed and could they send us job announcements. The alumni were also given a form on which they could request a job search from us. This last gesture of good will brought us good returns on our questionnaires, including newspaper clippings of job announcements and advice on seeking jobs.

Our alumni sympathized with the job problem and heralded the efforts to establish a library of potential employer addresses. When questionnaires were returned from alumni or potential employers, a data bank was established to include:

- employer's name and address
- degrees required
- frequency of hiring
- types of background needed
- areas of speciality
- summer or full time employment
- region of country

Students seeking employment could go to the placement office and request a job search, stating region of country, degree, and areas of specialty, and they would be provided with a list of addresses of potential employers.

Once a representative data base was established, it was possible to tally, using the FAMULUS glossary, the number of employers in each area of specialty. Furthermore, a student can access the program early in his career to determine the types of skills needed to get a job in his area of interest.

Obtaining and key-punching data soon became a large task, so student volunteers were sought to make the system work. Those who could type were taught key-punching and typed data sets. Others collected names of potential employers by checking journals, books, and newspapers, and others checked the alphabetized master list (also created by using FAMULUS) to see if employers were already listed in the file. One student was employed for five hours a week to maintain the computer files.

The result of developing this system has been an improved morale among the

students. Many are anxious to contribute their time. Long lists of addresses of potential employers are exciting. Interviews have resulted. The program is new and cannot really be evaluated in terms of its original objective, but it has helped in the counselling process. Our students are more aware of potential job sources and they have been able to explore possibilities unknown to them before. Perhaps most important is their realization that the faculty, who started the program, care.

"Zebra" Keys

A relatively recent use of information retrieval has been in the field of taxonomy. The first published use I can find was developed by undergraduate Larry Morse at Michigan State University (1968). Morse developed dichotomous computer keys and data matrices for use in undergraduate teaching and for taxonomic research on plants. Since then, others have applied the technique to bacteria (Bascomb, Lepage, Curtis, and Willcox 1973) and many additional plant uses (Beaman 1971a, Furlow & Beaman 1971, Ntarella 1971, Pankhurst 1971). The technique has not become widespread largely because of lack of supportive software for data matrices. Most would-be users simply won't implement a system which can only key *Carex*, *Acer*, Michigan dicot families, Michigan ferns, and a few others of similar limited scope. Nevertheless, certain uses of the programs for taxonomic information retrieval have been highly successful and deserve attention.

Before discussing my analysis of the virtues of information retrieval in taxonomy, I shall address the skeptics. The two major criticisms which seem to arise are cost and "what's wrong with ordinary keys?" To answer the first, I must cite the Michigan State University program, where use of computerized keys in the introductory plant taxonomy course cost only a few dollars per student (Morse 1974), about the same as the usual lab fee and considerably less than a hard-bound key. To the traditionalist who prefers ordinary hard-bound keys, the following paragraphs will discuss the many non-traditional uses not possible with ordinary keys.

It is the purpose of this paper to illustrate both traditional and non-traditional uses of identification programs, pointing out some of their under-utilized advantages. To this end I shall discuss the following purposes for their use:

1. aid identification
2. eliminate making uncertain choices
3. provide a list of possible remaining taxa for which one can check herbarium specimens and descriptions

4. increase powers of observation
5. permit location of taxa with unusual characteristics
6. permit self-testing of important characteristics to be used for separating higher taxa
7. permit review and testing of characteristics for each taxon, with immediate feedback, in preparation for field recognition
8. permit grouping of taxa with common characteristics and provide a capability for finding what characteristics are needed to separate them
9. permit teacher-generated tests on keying ability.

Although these programs were initially designed for purposes of identification by biologists at the undergraduate through Ph.D. level, their utility extends throughout the educational system. This will become evident with the non-traditional uses.

As already pointed out, the major limitation on use of computerized taxonomic identification keys is the lack of available data matrices. These are relatively easy to construct and computer software is available to check data sets for capability of separating taxa and for developing keys (Morse 1974). Figure 1 illustrates the stepwise process.

Perhaps the most difficult task is the choice of characteristics to use. This must be done first and requires anticipating all the characteristics which might be needed to separate any of the taxa. Some are always missed, so it is useful to organize information by sections and leave room for additional descriptors to be added. A program such as CHARLIST is used to number the descriptor couplets into sets of 80 to comply with standard computer column numbers. If new descriptor couplets are added, it is a simple matter to renumber these with the program. For small keys, such as the example provided herein for *Lemna*, data matrices are not difficult to construct. For large data sets, it is beneficial to break the data matrix into smaller matrices based on natural groupings, such as monocots or dicots, united or separate petals. Whenever possible, confusing characteristics should be avoided, or allowance should be made for several interpretations. For example, if one author considers an ovary to be inferior and another considers it perigynous, code the data to accept both.

Many sorts of code systems can be developed, but the following system was initiated by Morse for Michigan State University (Beaman 1971b), and I have used it to maintain compatibility. The members of a descriptor couplet are assigned a number 100 apart.

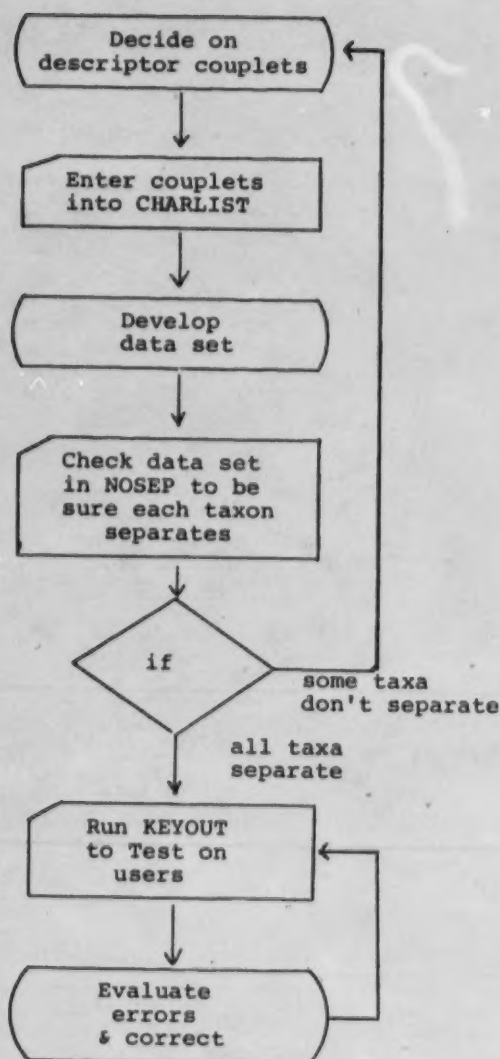


Figure 1. Steps in preparation of taxonomic data matrices.

The data matrix is coded in the data column matching the units and tens of the descriptor couplet number, as follows:

1--if only first characteristic of descriptor couplet is present

2--if both first and second descriptors apply

3--if only second characteristic of descriptor couplet is present

4--if neither applies

blank--if information applying to characteristic is unknown

A typical worksheet data matrix and set

of descriptor couplets are shown for Lemna in Tables 1 and 2.

Table 1. Sample descriptors and data code worksheet.

	roots: 001-yes; 101-no	roots: 002-(1); 102-(3)	size: 003-<2mm, granular; 103-larger	color: 004-purplish; 104-green	petioles: 005-present; 105-absent	nerves: 006-(1); 106-(3)	symmetry: 007-asymmetrical; 107-symmetrical	shape: 008-oblong; 108-rcund	punctate: 009-not punctate; 109-punctate	papilla: 010-terminal; 110-central	surface: 011-flat above; 111-rounded above
<u>Lemna minor</u>	1	1	3	3	3	3	3	3	1	4	1
<u>L. trisulca</u>	1	1	3	3	1	3	3	1	1	4	1
<u>L. minima</u>	1	1	3	3	3	1	3	3	1	4	1
<u>L. perpusilla</u>	1	1	3	3	3	3	1	1	1	4	1
<u>L. valdiviana</u>	1	1	3	3	3	1	1	2	1	4	1
<u>Spirodela polyrrhiza</u>	1	3	3	1	3	3	1	3	1	4	1
<u>Wolffia columbiana</u>	3		1	3	3		3	3	3	4	3
<u>W. punctate</u>	3		1	3	3		3	3	1	1	1
<u>W. papulifera</u>	3		1	3	3		3	3	1	3	1
<u>Wolffiella floridana</u>	3		3	3	3		1	1	1	2	1

1 = 001
3 = 101
2 = either applies
4 = neither applies

Table 2. Descriptor couplets for Lemnaceae.

001	Roots present
101	Roots absent
002	Roots one
102	Roots three
003	Size under 2 mm
103	Size greater than 2 mm
004	Underside of plant purplish
104	Underside of plant green
005	Petioles present
105	Petioles absent
006	Nerves one
106	Nerves three
007	Blade asymmetrical
107	Blade symmetrical
008	Blade oblong
108	Blade nearly round
009	Blade smooth
109	Blade punctate
010	Papilla (if present) terminal
110	Papilla in center
011	Blade flat above
111	Blade rounded above

Once the data set has been developed, a program such as NOSEP is used to determine whether each taxon has a unique set of characteristics in the data matrix. Once sufficient characteristics are supplied to the data matrix, the key must be tested by users. Students feel a kind of accomplishment when they discover errors in the key but they should be aware that you want them to look for errors. Finding actual errors seems to reinforce their feelings that they know what they are doing. Many systems provide a file for users to enter comments and suggestions when they have finished using the program, and this could be used as a reservoir for error records.

To use the KEYOUT program, a student should first be provided with the list of descriptor couplets such as the one provided in Table 2. Once he becomes familiar with the descriptors, the student must examine the specimen provided and decide which descriptors apply to his specimen. At this point, one of the advantages of a computer key arises. He is not forced to make an uncertain choice. He should list only those descriptors about which he is certain. Once he has made his decisions, he types the appropriate descriptor numbers into the computer. A list of remaining taxa will be provided by the program. At his request, he can also obtain a list of useful characteristics. This list contains only those descriptor numbers which will help separate the remaining taxa. Once he has provided sufficient information, the program will provide a suggested identification.

At this point, one can see two additional advantages. First, the list of useful characteristics will narrow the number of possible decisions. Second, if he cannot provide, with certainty, any additional descriptors, he has a short list of taxa which he can check by use of a herbarium or published taxonomic descriptions. Checking the remaining taxa is difficult with traditional printed keys because it requires following all remaining branches of the key to their conclusion to obtain the list.

The non-traditional uses of computerized identification keys, I believe, offer far-reaching educational possibilities. Among these, the capability of increasing powers of observation is to me the most exciting.

Most of our educational testing system is based on a stimulus-response type of recall. A teacher guides a student's learning by choosing those things which the student must learn. Let us compare this to letting nature be the teacher with the computer program as the evaluator.

As an example, let us have a student select a peppermint plant. He sits down with his plant, rolling it between his fingers. He notices two things as he

does. First, its stem is not round. In fact, upon inspection he finds that it is square. Second, it has a pleasant peppermint odor. These two observations are unusual to the student, and he is likely to remember them. But how much more can he notice?

A challenging use of the taxonomic keys in this regard is to ask the student to observe his plant first, noting every characteristic he thinks could possibly be important, forcing him to pay attention to detail. He will begin to compare the plant with others he has seen. Suggest that he try to describe the plant so well that he can identify it on the first try without further reference to the plant. For this use, a TV screen terminal which provides descriptor lists and allows him to choose by touch might be ideal, but a written list is adequate. When he provides the computer program with his list of descriptors, he will get immediate feedback as to how well he completed the list. If he is successful, the "game" is likely to draw him into friendly competition with classmates to see who can accurately and completely describe the most plants in the least time. If he is unsuccessful, he has a second chance to look at the plant and complete its description. His first few tries will probably not meet with success (ours usually did not), but soon he will become more observant and more aware of what is important. This "game" is applicable to awareness training in many fields of interest, including art, sounds, feel, people, ad infinitum.

This brings us to a closely allied non-traditional use: determining unusual characteristics. To the student twirling the peppermint plant between his fingers, the square stem is unusual. If he supplies the computer with that characteristic alone, he soon finds that many plants can have square stems. He can, as he examines additional plants, begin to recognize a characteristic that he has not encountered before and immediately find out how unusual this is.

The data matrix which has provided me the greatest teaching success for identifying unusual characteristics and most other uses in teaching taxonomy is one for characteristics of families of flowering plants found in the state. One concept one might wish to convey to students is the "gestalt" of a family. That is, what is its wholeness, what makes a family a family, and not merely a genus or other taxonomic group. To this end, a student must recognize on a plant which characteristics are important in classifying families. He can again test his powers of observation, but he must add to that a weighting factor whereby he selects those characteristics which are most important. The immediate feedback of the computer will facilitate such learning.

In this same regard, then, the computer program provides a means of self-testing and review. It provides an avenue of reinforcement when a teacher is not available. A student can go to the computer with a list of taxa he wishes to learn. As he goes through the list, he can try to provide the computer a list of descriptors which he has chosen based on his recollection of the organism. His object is to be able to describe his organism so as to completely separate it from all others. I have found this technique particularly useful for learning field characteristics. It is a replacement for the standard flash card method, but has the added advantage of having a broader data base and permitting the student to use his own observations of what are the most useful characteristics to remember. Using this method, he can remember that a plant has red stems, whereas I teach that it has white pith, and both characteristics will successfully separate it; he is not forced to use my device, but rather can use the characteristic which is more outstanding to him.

A word of caution is in order here. The data matrix is likely to contain errors, so a student should be encouraged to develop his concepts of useful characteristics with the organism at hand. This serves two purposes. He develops a mental image of the organism to back up his descriptor memory and he locates any discrepancies between his specimen and the data matrix. In this way, errors or deficiencies in the matrix can be located.

I have found that students often like to use the computer to find out what other organism might be confused with theirs. For example, if one approaches the study of plant families by trying to characterize them so that unknown members can be placed in a family when they are first encountered, a student will soon want to know where he is likely to make mistakes. He has noted that milkweeds have milky juice, but what else does? Once he obtains that list, what, then, separates the milkweed? Since milky juice is probably the first unusual characteristic he will notice on a new plant, it is important that he then have a mental sifting procedure for the families with milky juice. By reviewing the families on the computer-generated list he can try the characteristics he feels will separate members of this group. I found my students, as exam time neared, making family group lists for all sorts of characteristics I had given them to aid identification. This soon gives them a feeling for common and rare characteristics, and it is not unusual for them to learn a few more families than they were assigned.

Once the student gains familiarity with computerized keying, both the teacher and the student need a measure of

how well each of the above objectives has been met. This can be accomplished by providing the KEYOUT program with a test option. The teacher must provide the student with a set of numbered organisms. Concurrently, the computer program is provided with a list of the positions in the data matrix for the test plants. If the student indicates that he is taking a test on specimen #6, the program will check his final identification against the data and congratulate him on his correct answer. If he has not reached the correct identification, he has the option of trying again or requesting the answer. This permits a student to evaluate his keying when no teacher is around to verify his identifications.

Finally, a student is ready to generate his own data set. Students who have evaluated this assignment have commented that they had not realized how difficult it is to select useful descriptors. When they tried their keys on actual organisms, they found they could have used many more characteristics. Others, who made very thorough descriptor lists, commented on the amount of research necessary to complete all the data for all the species. They began to empathize with the way a taxonomist must work as he identifies or classifies organisms.

The development of data sets by students should definitely be encouraged. It is an excellent learning experience which provides much needed data sets for additional keys. Students learn the parallel structure for setting up couplets and are forced to organize information. Assignments should, to be most effective, have three considerations. First, the key should be for a small, defined group. I find 8-12 is usually sufficiently complex without being mind-boggling for a beginner. Second, choose a genus, a section of a genus, a subfamily, or some other entity which is complete, and therefore usable. It should not merely be a key to the taxa "learned in this course." Students are most conscientious and interested if it will be used by someone. Third, the student should choose a group of organisms in which he is interested, or with which he has particular problems and wants to gain greater familiarity. When these criteria have been used, I have found students enthusiastic, and they have produced good, usable data sets.

In conclusion, the possibilities for using information retrieval for education are limited only by our imaginations. They have the potential of systematically training us to be more aware, and as a result, we can more fully enjoy our ecosphere.

Literature Cited

- Bascomb, S., S. P. Lapage, M. A. Curtis and W. R. Willcox. 1973. Identification of bacteria by computer:

- Identification of reference strains
J. Gen. Microbiol. 77:291-315.
- Beaman, J. H. 1971a. Some applications for the taxonomic data matrix: Six term papers by students at Michigan State University. Fl. N. Amer. Rep. 63. 103 pp.
- _____. 1971b. Introductory plant systematics: Lecture syllabus and laboratory manual, 4th ed. Dept. Bot., Michigan State Univ., East Lansing. 113 pp.
- Furlow, J. J., and J. H. Beaman. 1971. Sample taxonomic data matrices for vascular plants prepared by students at Michigan State University. Fl. N. Amer. Rep. 56. 118 pp.
- Morse, L. E. 1974. Computer programs for specimen identification, key construction and description printing using taxonomic data matrices. Publ. Mus., Mich. State Univ. Biol. Ser. 5(1): 128 pp.
- Natarella, N. J. 1971. A computer data package for the selection of ornamental shrubs. Fl. N. Amer. Rep. 63:33-42.
- Pankhurst, R. J. 1971. Botanical keys generated by computer. Watsonia 8:357-368.

USE OF THE PERSONAL COMPUTER AS
AN IN CLASS TEACHING TOOL IN
INTRODUCTORY BUSINESS STATISTICS

SIDNEY P. SYTSMA
SCHOOL OF BUSINESS
FERRIS STATE COLLEGE

ABSTRACT

The microcomputer has brought to the statistics classroom a substantial amount of low cost computer power. Many potential applications exist to assist the student in learning statistical concepts. At Ferris State College we have used a Radio Shack TRS-80 microcomputer as an in class teaching tool since September 1978 in Business Statistics. The microcomputer was used for three different purposes in the course:

1. As a graphics display device in the classroom.
2. As an in class problem solving tool for solving statistical problems.
3. As a generator of data and problems for class solution.

Evaluation of the results of the system after one year of use suggests cautious optimism as to the benefits of the system. However, many technical problems remain to be solved.

The Need for Microcomputer Based Systems

Introductory statistics education has changed conceptionally in the last few years from a computationally oriented course in which a great deal of time was spent performing calculations on rotary calculators, to a course in which the emphasis is on the selection of the appropriate statistical technique or techniques from a menu of available techniques to solve a specific problem. This has become feasible largely because of the availability of time-sharing computer systems and statistical software which is relatively easy to use.

When the statistics instructor focuses his instructional objectives on technique selection, solution sensitivity, and meeting technique assumptions as opposed to teaching computational skills, he soon discovers the need for a moderate amount of computer power in the classroom. Most frequently the required computer power is supplied by a central time-sharing system. There are at least two instances when this is not practical or possible:

1. There is no time-sharing system available or the existing time-sharing system cannot be wired to the classroom used.
2. The time-sharing system does not have sufficient visibility to be seen easily by the class.

A microcomputer based statistical analysis system can provide sufficient computer power to solve small to moderate sized statistical problems at low cost. Many microcomputers can be easily coupled to standard 70 ohm video monitors or video

projection systems for classroom use. Most microcomputers can be programmed in extended BASIC with relative ease. Many also provide easy to use graphics and plotting capabilities to provide plots of equations, histograms, probability distributions, regression residuals etc.

Considerations in Selecting a Classroom Microcomputer System

1. The brand selected must be a survivor. Many brands currently being marketed will no longer be in existence in twelve months.
2. The brand selected must be reliable with readily available service.
3. A high level (extended) version of BASIC must be available.
4. The microcomputer should have adequate memory (RAM). After one year, we feel 32K bytes is minimum.
5. The system should have at least one disk and a disk operating system.
6. The video display characters should be large enough to insure easy visibility by the class. It is much easier for a class of 30 to see a display of 16 lines of 32 characters than a display of 24 lines of 80 characters.
7. At least some graphics capability is necessary. Graphics capability puts life into the classroom. In general the higher the graphics resolution, the better. Minimum would be the Radio Shack with 48 x 128 characters on the screen.

Acquisition of Statistical Software

A great deal of statistical software can be obtained from outside sources such as computer user's groups which will run with minor modifications on most microcomputers having extended BASIC. Other sources include various software publications. Three of these are listed in the references. Much of the software from both of these sources was developed for early time-sharing systems and is neither interactive nor does it take advantage of the graphics capabilities of the modern microcomputer. The algorithms used are generally proven, and are adequate for classroom use. A useful route to follow in accumulating software for an instructional statistical system is to start with someone else's proven algorithm, and then modify it to be interactive, self instructing, and adequate graphically.

After examining what is currently available for statistical analysis in the microcomputer area, we feel that we can probably develop better statistical software than what is being marketed. The chief reason for this is that what we develop is specifically designed to take advantage of particular microcomputer features.

The Experiment: System and Software

We have used a Radio Shack TRS-80, with 2 disks, and 48K memory in the classroom for one school year. During this year we have written and tested software for use primarily in the classroom. We now have software available to perform most of the common statistical hypothesis tests for means and proportions, ANOVA, χ^2 tests as well as programs to perform simple and multiple correlation and regression plus confidence intervals and descriptive statistical programs. We are writing problem generation software that will provide controllable numeric data for student analysis, as well as the appropriate algorithm for problem solution. This will permit providing each student with his own individualized problems for solution. The system is operational in the classroom about 50% of all class periods.

The Experiment: In Class and Out of Class Usage

We use the system in the classroom to analyze data in a manner very similar to that used with a typical time-sharing system. The emphasis is on analysis sensitivity, and meeting statistical model assumptions. We can easily reanalyze data to examine the effect of changing a single observation. Particularly in the area of small sample hypothesis testing and the area of

regression the student can get a quick intuitive feel as to the effect on the analysis that changing as little as a single observation can have, and how that change impacts the analysis results.

We allowed limited student use of the classroom microcomputer during this academic year. The students could utilize the same programs demonstrated in the classroom. This proved to be quite popular with the students. We did have difficulty with the students handling the floppy diskettes. Diskettes are very fragile and sensitive. College students tend to be somewhat careless when dealing with equipment, and the dirt from their hands, the chalk dust, and the flexing and bending of the diskettes caused continual disk read problems. Although we plan to continue using the personal computer in the classroom, we feel that it is probably more feasible to have identical programs on the local time-sharing system for student use. Most of our programs will be running on the time-sharing system by September. An alternate solution would be to seal the appropriate diskette in the disk drive so the student could not remove it. Since we were limited to a 2 drive system, and our statistics package was on three different diskettes, this would have been difficult without eliminating some programs.

The Experiment: Outcomes

There is no question that the students using the system liked it. In the three terms that I have used it, my student evaluations indicate that it one of the most positive areas in the course. In particular they like the emphasis on statistical reasoning with the computer as the analysis tool. In addition, those students who have used the problem generation programs generally feel that they are interesting and worthwhile.

The decision to modify course objectives to deemphasize computational aspects and to emphasize statistical reasoning, technique selection, and computer analysis was made 18 months ago. During this 18 month period the course has been taught three times with the system, and three times without. Although the lack of a preplanned experimental design tends to obscure any conclusive comparisons, a simplistic examination of the data seems to show improvement on objectives related to selection of the appropriate statistical technique and the interpretation of the results of the analysis. The same data tends to indicate about the same level of student performance on test items which measure computational skills.

152 Conclusions and the Future

The system works well enough to continue using it in the classroom, and to continue to develop additional software for classroom use. In addition we have proposed adding an additional computer system to eliminate some of the transportation problems of having only a single system to work with. In addition we are converting the existing software to the timesharing system. Any new software that we develop will be designed to operate on both the campus timesharing system as well as the micro-computer system. This allows the instructor to demonstrate the use of the package in the classroom.

REFERENCES

- 1 Alonso, Joseph R., Simple: A Software Handbook of Statistical Techniques (Manhaca, Texas, Stirling Swift Publishing Co., 1978)
- 2 Poole, L. and Borchers, M., Some Common BASIC Programs (Berkeley, California, Adam Osborne & Associates, Inc., 1977)
- 3 Brown, R. W., BASIC Software Library Vol. II (Key Biscayne, Florida, Scientific Research Inc.)

USING THE COMPUTER TO FOSTER CREATIVE INTERACTION AMONG STUDENTS

Sharon Dugdale
Director, PLATO Mathematics Project
Computer-based Education Research Laboratory
University of Illinois

ABSTRACT

The network characteristics of the PLATO computer-based education system permit students to communicate not just with the computer but with each other. This capability can be exploited in educationally significant ways. In addition to the social interaction which occurs when students help each other at the terminal, and when they work together at a terminal in two-student lessons, students can interact in "real time" from widely separated terminals, and they can see and learn from products generated by other students who were working not only at other places but at other times. This results in a variety of experience, a richness of expression, and a degree of motivation that is difficult or impossible to attain when a student works at a terminal in effective isolation from his peers.

THE COMPUTER SYSTEM

The PLATO computer-based teaching system,^(6,7) developed at the Computer-based Education Research Laboratory (CERL) at the University of Illinois, consists of a large central computer connected to about 1000 terminals located in public schools, community colleges, universities, correctional institutions, military training schools, and other institutions throughout Illinois and other states. Each terminal consists of a screen on which the computer displays text and graphics to the student and a keyset which the student uses to interact with the computer. Most PLATO terminals also allow students to communicate with the computer by touching the screen. The system allows graphic displays, powerful answer-judging algorithms, and "instant" response to student input. Instructional materials are available in over 100 subject areas, ranging from lower elementary grades to university graduate course level.⁽⁵⁾

PLATO IN THE SCHOOLS

Since 1974, PLATO terminals have been a part of the learning resources in six to twelve intermediate grade classes in the Champaign and Urbana, Illinois, public schools, where the students use PLATO as a part of their mathematics work. A typical classroom is equipped with four PLATO terminals. The classroom teacher is responsible for scheduling the use of the terminals and for assigning PLATO topics so as to meet the specific needs of his students. On-line and hard copy data about students' progress and performance are available to the teacher. An overview of the use of PLATO in school mathematics is available in Dugdale and Kibbey.⁽²⁾ For more information about the structure of the curriculum,

see Dugdale and Vogel.⁽⁴⁾ Results of a two-year external evaluation by Educational Testing Service of Princeton, New Jersey, indicated that these PLATO materials had a marked positive influence on students' achievement in mathematics and attitude toward math.⁽⁸⁾

INTERACTION AMONG STUDENTS

Terminals are grouped together in the classrooms in order to encourage interaction among the students. A student will often be proud of something he has done and will be eager to show it or explain it to a friend. Other times a student may turn to a classmate for help with a lesson. Generally, the materials are so motivating that the interaction among students at the terminals is centered around their work and does not take them "off task". A student who needs help is rarely willing to let another student take over and do the work. Usually the student will accept only the help he needs in order to become self-sufficient again.

Because tasks are generated for each student individually, the students also find that it is not helpful to copy an answer from a friend's screen. They can, however, learn ideas and strategies from watching a friend. For example, two students are working at adjacent terminals. PLATO displays some balloons tied to a number line.* The task is to tell where on the number

*The lessons that are discussed in this paper are part of a curriculum designed by Sharon Dugdale and David Kibbey.⁽³⁾ Other lessons intended to promote interaction among students have also been developed at CERL.⁽⁹⁾

line to shoot darts to burst the balloons. (See Figure 1.) When a student types a number, a dart flies across the screen and lodges in the number line at the point corresponding to the student's answer. One student, unsure of what to do, looks to see what his friend is doing. (On PLATO, as elsewhere, watching a game being played often helps one to understand the directions.) His friend has just shot a dart at $3/4$ and hit a balloon. So the student turns back to his own terminal and types $3/4$. But the dart does not hit a balloon, because there is no balloon at $3/4$ on his number line. Clearly, copying his friend's answers does not help. So he turns to his friend's terminal again, but this time the interest is in learning how to locate numbers on the number line. After a few minutes of watching and a few more trial darts, he is able to progress through the lesson on his own.

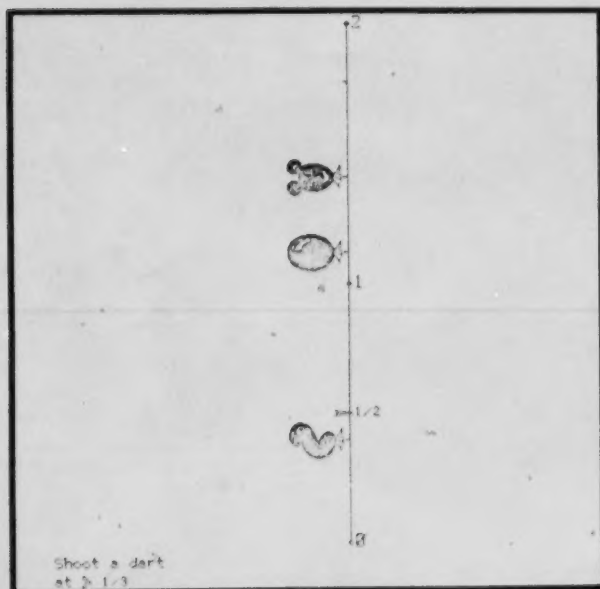


Figure 1. A display from lesson "Darts". Balloons are tied to the number line. The student shoots darts by saying where on the number line the dart should go. The dart flies across the screen and lodges in the number line, popping a balloon if it hits one. In this display the student has already shot a dart at $1/2$ and has chosen to shoot the next dart at $1/3$.

LESSONS THAT ENCOURAGE STUDENT INTERACTION

One-terminal Lessons

One kind of PLATO lesson designed to encourage interaction among students is that intended for use by more than one student at the same terminal. Such lessons are usually "games" for two players. Sometimes the students simply compete in an activity such as "Fractions Basketball",⁽³⁾ where the players shoot baskets by writing appropriate fractions. Other lessons have students make up problems for each other, as in "High Wire",⁽³⁾ in which one student "hides" a monkey on a number line by writing an addition of

fractions problem, then the other student tries to locate the monkey by writing the correct sum.

Interterminal Lessons

A broader interaction and sharing of ideas is possible by exploiting the network characteristics of the PLATO system. One popular interterminal lesson is "Torpedo", a game that provides practice in estimating distances on a number line. (See Figure 2.) One player controls a boat, and the other a submarine. The object is to move to a point above (or below) the opponent's piece and shoot it. Sea animals are obstacles that can be used to hide behind. The players take turns. On each turn a player can move his piece, or he can move-and-shoot.

Because each game lasts only a minute or two, the student can play with many different opponents in a short time. The opponent's move specifications are shown on the screen, so the student is likely to see a variety of methods, e.g., common fractions, decimals, or expressions with operations.

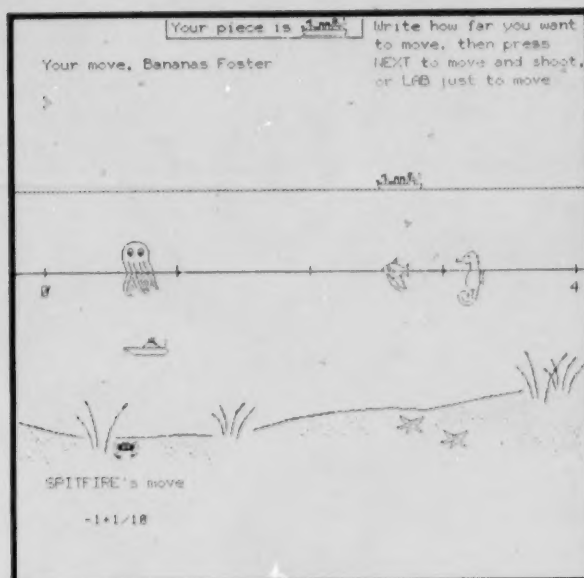


Figure 2. Display from lesson "Torpedo". The student has chosen to call himself "Bananas Foster". His object is to move his battleship to a position above his opponent's submarine and shoot the sub, avoiding the sea animals in between. To do this he must read and use the scale on the given number line. The opponent, "SPITFIRE", has just moved his submarine " $-1+1/10$ " to hide under the octopus. (One might speculate that SPITFIRE's thinking could have been that a move of -1 would be a little too far to the left, so adding $1/10$ would be about right.) Because each player sees the expressions his opponents write, students quickly learn a variety of useful ways to write such expressions.

A simple example of the resulting exchange of ideas is the students' use of zero. Typically, a student who is new to this lesson will not use zero. Even when he finds his game piece

positioned ready to shoot the opponent's piece, the student will enter a small non-zero number rather than zero. Sometimes, much to the student's distress, the small non-zero move will not only miss the target, but will leave the student's piece where a more experienced opponent will shoot it with a move of zero. It seldom takes more than one or two such experiences before the new player realizes that zero is a perfectly good number which measures a sometimes-useful distance.

"Torpedo" allows students to interact with many different people, but each short game involves only two students. Another interterminal lesson, "Obstacle Course", allows as many as four students to participate together at a time. (See Figure 3.)*

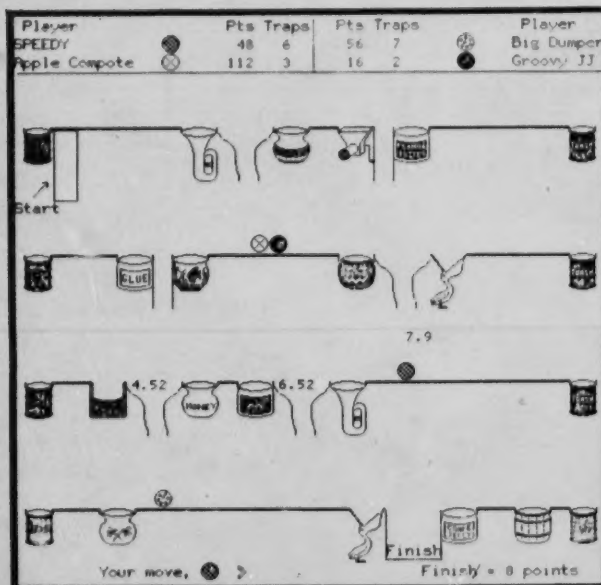


Figure 3. Display from lesson "Obstacle Course", which provides practice in quickly estimating differences between decimal numbers. Four students, playing from separate terminals, have chosen to call themselves "SPEEDY", "Apple Compote", "Big Dumper", and "Groovy JJ". Each player has a ball on the obstacle course. The object is to move one's ball through the course via the chutes without falling into any traps. Points are earned by reaching "FINISH". This is the display on SPEEDY's terminal. SPEEDY's ball is at point 7.9 on the third line. SPEEDY has to write a number to move his ball to a chute through to the next line. These wide chutes are easier to hit with a quick estimate. The narrow ones (see lines one and two) require more accuracy. Players do not wait to take turns. Each makes successive moves as quickly as he can.

The several number line activities described here may appear at first glance to be undesirably similar to each other. A reader with this perception should look more carefully. Each activity has a different objective and makes different demands on the student.

A feature of both "Torpedo" and "Obstacle Course" is that students make up pseudonyms for themselves. (Students are free to use their real names, but they almost never do.) Besides being considered fun, this anonymity avoids embarrassment over mistakes and encourages participation of the students most in need of the practice.

Library Lessons

Another type of lesson that uses the network capabilities of the PLATO system is the "library" lesson.** In a library lesson a student does some mathematically relevant creative work and is then given the option of saving his work in a "library" so that other students can see it. He may replace his work with something new at any time. Each class has a separate library. Students are able to see the work stored in other classes' libraries, as well as their own. This provides an ongoing exchange of ideas among students in many different classrooms, even in different schools and communities.

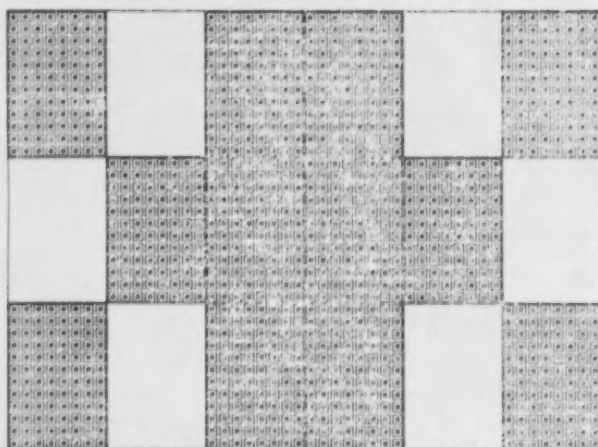
In the "Paintings Library" the student is asked to "paint" a particular fraction of a box. By touching the screen, he can draw lines on the box and paint areas of the box. When he has finished painting the correct fraction of the box, the student can choose to save his painting in the library. Figure 4 shows a few such student paintings.

Social interaction is quite apparent when fads spread through class libraries. For example, one student might paint a fraction of the box to show his initials. Within a few days, there will be many paintings showing other students' names and initials. Our experience has been that a large number of students participating in a library is a very important factor in its success. This lesson was first tried with only four students, and although they were described as bright, creative students from an open classroom environment, their paintings were quite unimaginative. Each student did some paintings and looked at the paintings done by the other three, but new ideas were few and enthusiasm was only mild. However, during a three-week period when 300 students used the lesson, it exploded with new ideas as students looked at each other's work and responded with their own.

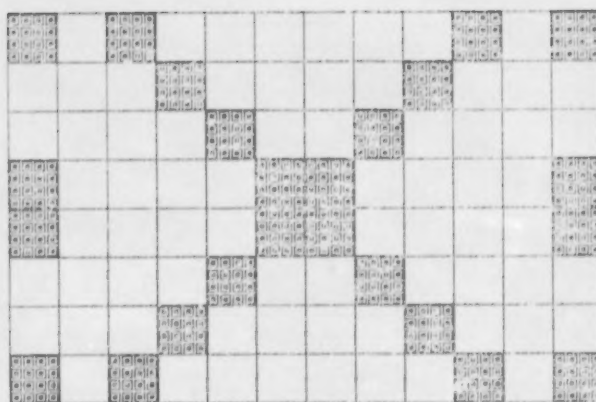
This lesson and others like it provide strong motivation for students to engage in tasks which require them to relate mathematical concepts to concrete models. Students are observed to engage in many lines of mathematical thought when creating a painting. For example, if the student has chosen $\frac{3}{5}$ as the fraction to represent, and

**The idea of library lessons was originated by David Kibbey. The work was begun in 1973 and first reported by Robert B. Davis, then Director of the PLATO Mathematics Project, at the NATO Advanced Study Institute on Machine Representations of Knowledge Conference, University of California, Santa Cruz, California, July, 1975. The resulting paper was published in *Machine Representations of Knowledge*.⁽¹⁾

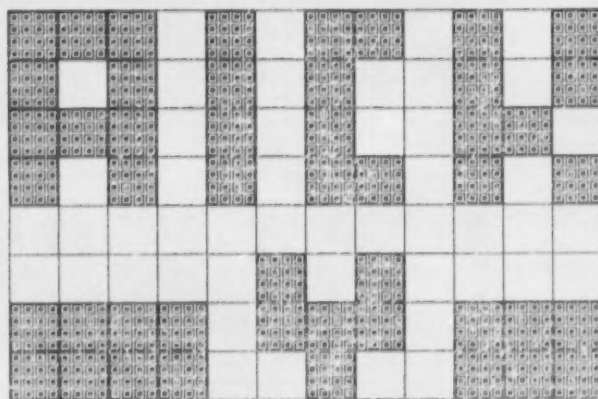
This is how mara painted $\frac{2}{3}$ of the box.



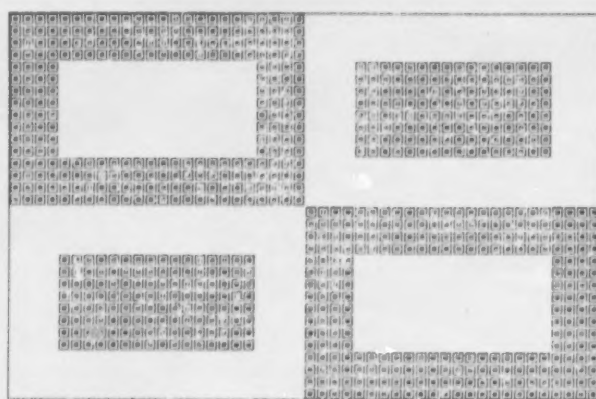
This is how johanna painted $\frac{1}{4}$ of the box.



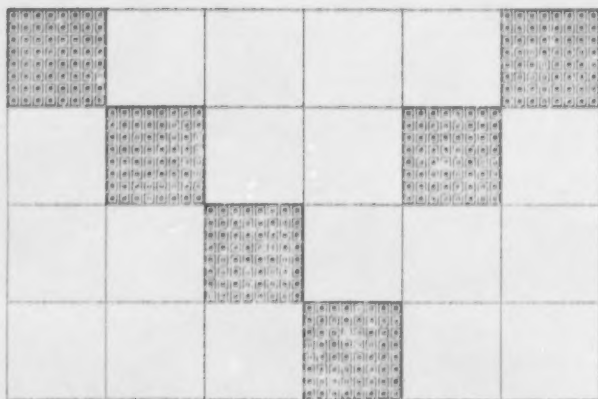
This is how ricky painted $\frac{1}{2}$ of the box.



This is how brian painted $\frac{1}{2}$ of the box.



This is how leah painted $\frac{1}{4}$ of the box.



This is how jim painted $\frac{1}{2}$ of the box.

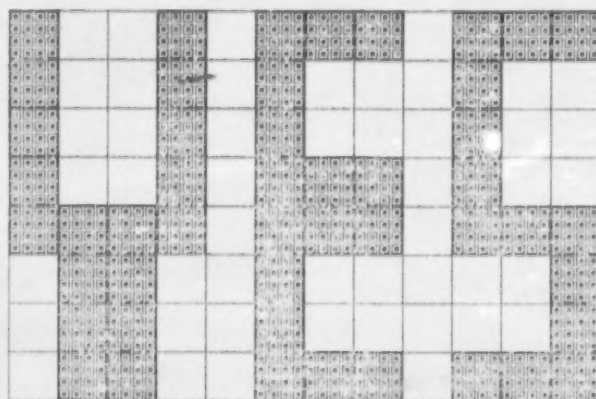


Figure 4. Some paintings that students have chosen to share through the library. The student's name and the fraction painted appear above each painting.

in doing a creative pattern has mistakenly painted $1/20$ too much, PLATO will say, "You have painted $13/20$ of the box". The student must then decide whether that is too much or too little and how much paint to add or remove. Some students will count same-size pieces and possibly use equivalent fractions. Others will visualize moving painted parts around to make an equivalent simpler painting. Still others may partition the box and deal with the parts separately, deciding whether $3/5$ of each part is painted.

Later in the curriculum is the "Paint Addition Library". Here the student can use two kinds of paint to illustrate an addition problem. Students can ask PLATO for a problem to illustrate or make one of their own. In either case, when the student has correctly represented the problem, he must tell how much of the box is painted all together. This may be done by several methods, including drawing lines on the box until it is

divided into some number of equal pieces, each of which is either all painted or all unpainted. (This is a visual method of finding a common denominator for the two fractions.) Figure 5 shows four students' representations of various additions. 157

"Skywriting and Spider Web" is a library lesson that lets the student write a "program" to move an airplane or a spider around the screen. (See Figure 6.) The airplane leaves a trail of smoke, and the spider leaves a web, so that the student's program makes a pattern on the screen. The simple programming language has three commands:

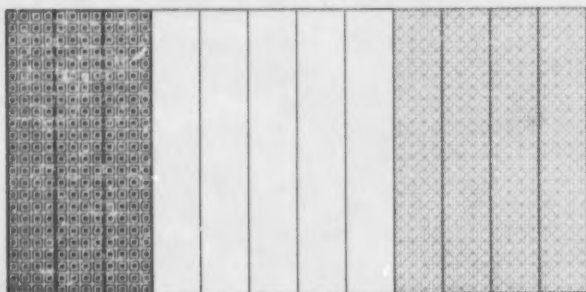
"f" to go forward (specify how many units)

"t" to turn (specify what fraction of a turn)

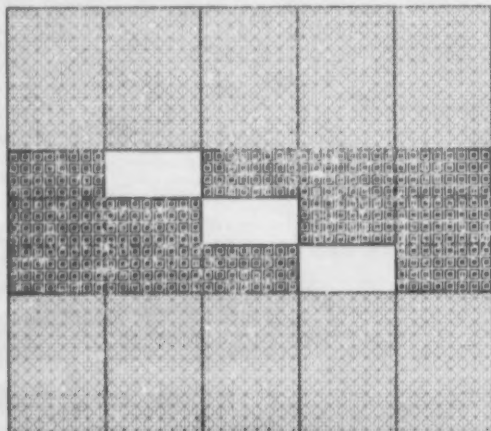
"r" to repeat the program

There is also an option to turn the smoke or web off and on.

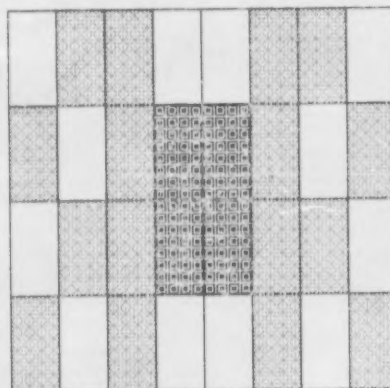
This is eric's painting: $1/4 + 1/3 = 7/12$



This is benson's painting: $4/15 + 2/3 = 14/15$



This is lisa's painting: $4/32 + 16/32 = 20/32$



This is allison's painting: $48/64 + 24/64 = 1$

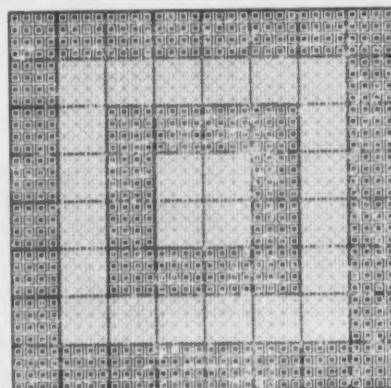


Figure 5. Four students' contributions to the "Paint Addition Library".

When the student has finished writing a program, he has the option of giving the resulting picture a title and saving it in the library. Figure 7 shows some programs from the library. In each frame the student's program appears at the top of the screen. At the bottom of the screen is the student's name and the title he has chosen for the picture. Each student has his own special space in the library to save a program. He may replace his program with a new one whenever he chooses.

This lesson gives the student experience with a simple programming language, with iteration, and with the idea of a correspondence between symbols and pictures. In addition, it applies fractions to the "turn", a fraction model very different from the common ones such as regions and discrete sets. There is also a great wealth of geometric content that can be pursued. For example, the spider's turns are spot rotations, while the airplane's turns are arcs of circles. One interesting exercise is to explore the difference this makes by running the same program with the spider, then with the airplane. Another possibility is to explore various "classes" of programs; e.g., programs of the form "n forward, 1/m turn, repeat" always make a regular polygon with m sides of length n.

CONCLUSION

The network characteristics of the PLATO computer system allow students to communicate not only with the computer, but with each other. This can have large educational payoffs. Students generate and share a wide variety of approaches to what is traditionally very stereotyped subject matter. The richness of the exchange is enhanced by the fact that the sharing takes place not merely across students, but across classes, schools, and even communities. Far from being a mere drill-and-practice robot, the computer can be a medium for social interaction, mathematical discovery, and aesthetic expression.

REFERENCES

1. Davis, R. B., S. Dugdale, D. Kibbey, and C. Weaver, "Representing Knowledge About Mathematics for Computer-Aided Teaching: Part II — The Diversity of Roles that a Computer Can Play in Assisting Learning", Machine Representations of Knowledge (eds. E. W. Elcock and D. Michie). Dordrecht: D. Reidel Publishing Company, 1977.
2. Dugdale, S., and Kibbey, D., Elementary Mathematics with PLATO, second edition, July 1977, CERL Publications, University of Illinois, Urbana, Illinois. Support for this curriculum development was provided by the National Science Foundation (US NSF C-723) and by the State of Illinois.
3. Dugdale, S., and Kibbey, D., The Fractions Curriculum of the PLATO Elementary Mathematics Project, March 1975, CERL Publications, University of Illinois, Urbana, Illinois.
4. Dugdale, S., and Vogel, P., "PLATO Instruction for the Hearing Impaired", Proceedings of the Second International Learning Technology Congress and Exposition of the Society for Applied Learning Technology, February 1978, Orlando, Florida.
5. Lyman, Elisabeth R., PLATO Curricular Materials, Number 7, February 1979, CERL Publications, University of Illinois, Urbana, Illinois.
6. Lyman, Elisabeth R., PLATO Highlights, March 1978, CERL Publications, University of Illinois, Urbana, Illinois. Other PLATO systems are based on computers located in Minnesota, Florida, Delaware, Quebec, and Belgium.
7. Smith, S. G., and Sherwood, B. A., "Educational Uses of the PLATO Computer System", Science, April 23, 1976, pp. 344-352.
8. Swinton, S., "Outcomes of the PLATO Elementary Demonstration", paper presented at the 1978 American Educational Research Association Conference, Toronto, Canada (March 1978).
9. Weaver, C. S., and Seiler, B. A., "Computer Assistance in the Social Processes of Learning", Proceedings of the Association for the Development of Computer-based Instruction Systems, Wilmington, Delaware, February 1977.



Figure 6. Writing a program in the "Skywriting and Spider Web" library. The airplane has been instructed to go seven units forward, then turn $2/3$ of a full circle. The program is recorded at the top of the screen as it is written. The key to the simple programming language is shown at the bottom of the screen.

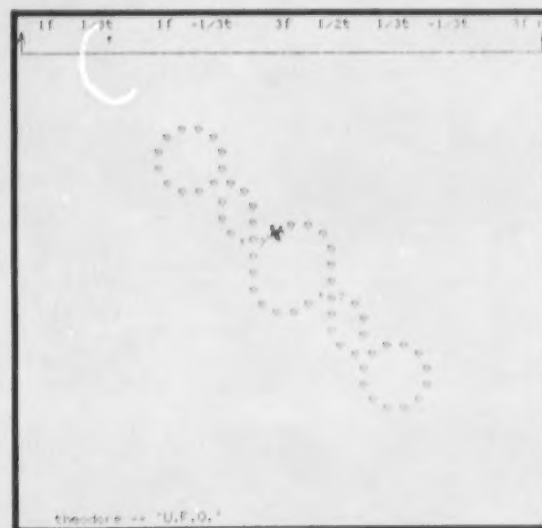
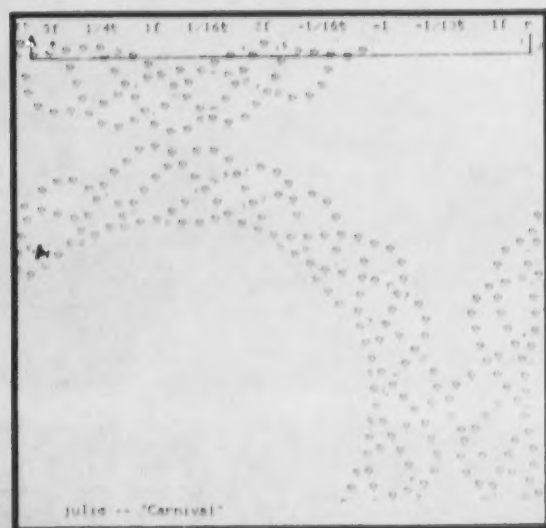
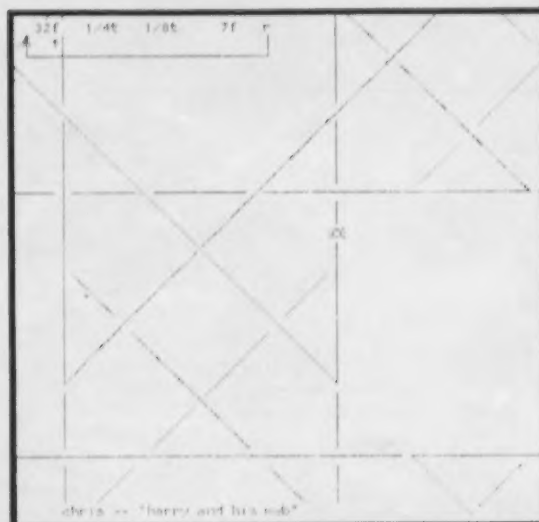
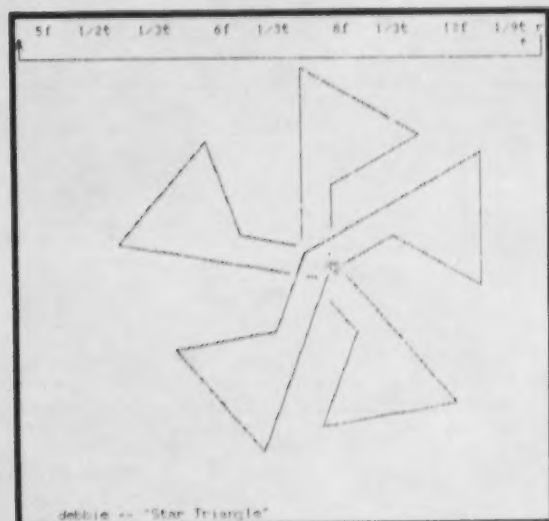
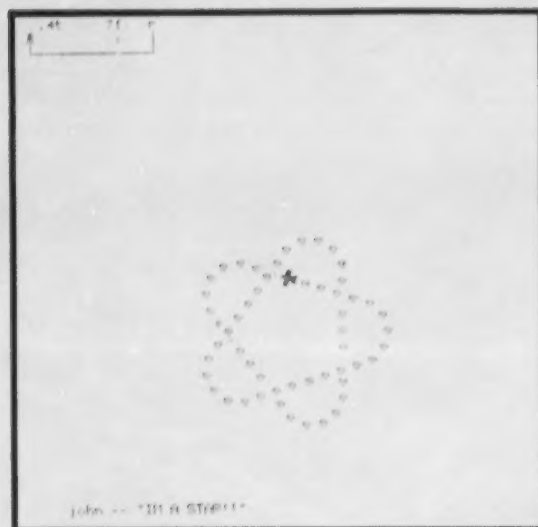
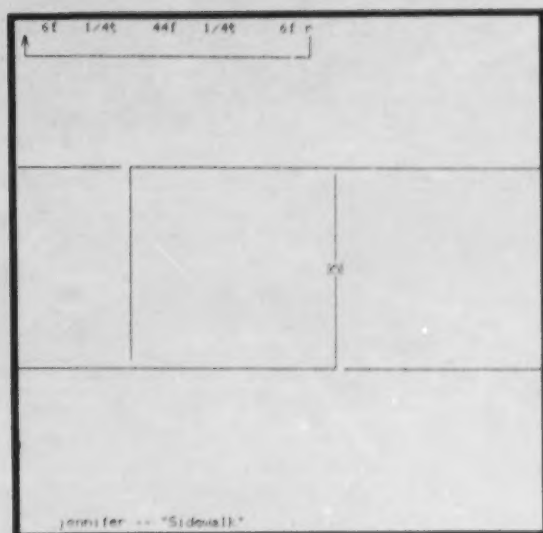


Figure 7. A few students' programs from the "Skywriting and Spider Web" library.

USES OF COMPUTER SIMULATIONS IN INTRODUCTORY PSYCHOLOGY

Douglas B. Eamon and William N. Hayes
Albion College

Generating understanding and appreciation of the methods and procedures of scientific research is an important objective of many introductory psychology courses. Space, equipment, and time limitations make conducting actual experimental research impossible. Computer simulations of actual investigations allow the introductory student to simulate designing and conducting their own research. Several simulation programs currently used at Albion College are discussed, and data concerning student use and student ratings of the programs are presented.

This paper presents an outline of the uses of computer simulations of laboratory experiments in the introductory psychology course at Albion College. Computer simulations of experimental research in psychology have existed for several years. Few of the programs available, however, are designed for use at the introductory level. We feel our implementation of computer simulations in the introductory course represents an unusual, if not unique, step in pedagogy at this level.

A general introduction to the rationale underlying the use of computer simulations in this course will be followed by a description of the programs we have used in our department. We will also present summary statistics regarding student evaluations of the simulations.

Rationale

Virtually all introductory psychology courses include in the first few weeks a brief explanation of the nature of psychology as a scientific discipline. Along with this an explanation of how psychologists go about collecting facts and verifying theories is also undertaken. For most areas in psychology, this method is the experiment. The experimental method and substantive content should thus become inseparably fused in this approach.

Experiments in psychology, however, are a costly and time-consuming business. Large numbers of human or animal subjects are often required, frequently meeting specific characteristics. For example, an experiment investigating the effectiveness of various forms of treatment of mental illness may require that all subjects be psychotic; an experiment in learning may require that subjects fall within certain IQ ranges; an experiment in neurophysiology may require all animals be prepared with specific types of brain lesions.

Because control of these and many other

variables is often impractical or impossible for experiments in an introductory course, most instructors sacrifice the requirement that the student perform experiments beyond those of the most trivial type. Those that are performed usually amount to not very elaborate demonstrations rather than the challenging exercises in creative thinking, decision making, data analysis, and deduction which characterize research by professionals.

Even if suitable subject populations could be obtained, other factors severely limit the types of experiments which could be performed by students. Expensive equipment, large investments of time and energy, and other logistic and practical problems have the effect of eliminating many of the most interesting and important experiments from consideration.

The use of computer simulations allows the instructor to bypass these problems in experimental investigations, yet retain many of those aspects which are most directly associated with the course objectives.

A variety of experimental simulations in psychology are currently available. Most of these allow the instructor or the student - or both in consultation - to select a topic from among several content areas, suggest an hypothesis, and then design an experiment which will test the hypothesis.

The specific steps involved are usually the following:

1. The instructor assigns (or the student independently selects from a prepared list) a topic area of interest. Normally, this area would be relevant to some content area under discussion in the class.
2. The student explores background information relevant to the topic and selects from among several specified potential independent

variables one which he or she would like to investigate. In the course of this exploration a portion of the relevant literature is examined, and the student acquires factual and theoretical information of a substantive type.

3. On this basis the student designs an experiment manipulating the independent variables selected and controls for secondary variables which may affect the outcome of the experiment.

4. The student runs the computer simulation. The simulation program requires that the student fully describe the experiment he or she has designed. The program generates data which correspond to what might actually happen should the experiment be run using real subjects. The data produced are generated according to the specifications of the simulation programmer.

5. The student examines the data and draws conclusions relevant to the hypothesis. These are written up as a report and evaluated by the course instructor.

Several observations regarding the pedagogical value of this approach can be made. The instructor is able to effectively incorporate practical experience in scientific thinking and research methodology into content areas relevant to the course. In order to be able to run a simulation, the student must be familiar with the basic concepts and terminology associated with its content.

Students can investigate problems which are closely related to their individual interests and needs. The motivational effect of this aspect of simulation can be illustrated by contrasting it with the traditional classroom demonstration/experiment, in which the student is often a passive data collector with little understanding, appreciation of, or interest in the activity.

The investment of both instructor's and student's time is small compared to "actual" experiments. Since the programs are normally run outside of class, the instructor is free to treat the content related to the simulation in greater depth during class, rather than supervising the data collection activities of the students.

As a result of the minimal amount of time needed to run a simulation, most students run more than one per terminal session. This encourages the development of a program of research or a set of several related experiments focused on the discovery of general principles rather than the identification of isolated facts.

Because each student conducts experiments of his or her own design, communication among students is facilitated. Students enjoy telling each other about what they have discovered, particularly if they feel their classmates may be unaware of their results.

Several simulation programs keep records of students' activities. An instructor is thus able to monitor the progress of each student by noting

such factors as the number of successful experiments performed, the number of failures, time spent at the terminal per run, etc. This information is useful in spotting the student who seems to be having problems in designing and running the experiments. Individualized help can be directed toward the solution of these problems.

Many programs utilize a "gaming" approach as an additional motivating aspect of the exercises. A student might be assigned at the beginning of the term an amount of money to be used in conducting experiments. A charge is levied for each subject run, each variable manipulated, etc. This discourages "shot-gun" types of experiments, and forces the student to think about such practical issues as the number of subjects needed to produce a measurable effect, the advisability of controlling all possible variables, etc. By carefully adjusting the gaming parameters, the instructor may make it possible for the student to actually make money with well-designed experiments.

Some simulations provide an evaluation of the students' experimental designs. For example, if more than one variable was manipulated in a two-group design, the program provides immediate feedback to this effect, and the student has the opportunity to correct the deficit without waiting to see the instructor for an analysis of the error.

In addition to these advantages, there is an important side benefit: the students' experience with the computer in running the simulations can reduce their anxiety about these tools. Indeed, some become interested in the programs themselves and how they work, and go on to take advanced courses in computer programming.

Programs used in introductory psychology at Albion College

The introductory psychology course at Albion College makes use of two sets of simulation programs.

The first set consists of ten stand-alone programs; each program is initiated with a separate run command. The programs include simulations of classical and operant conditioning, approach-avoidance conflicts, cortical lesions, concept formation, attitude change, and others. The second set of programs, called LABSIM, is a modular system; all modules are accessed within the program itself. This system will be discussed following our examination of the stand-alone programs.

Stand-Alone Programs

Written in BASIC by David Johnson during 1975 and 1976, these programs range in length from about 100 to 350 statements. The programs share a unique feature; that is, the student can request background information in the content area while at the terminal. The several paragraphs of information printed out include a brief history of the topic and definitions of important terms. This is followed by a set of questions about what the student would like to investigate. For example, in the simulation of classical conditioning the

student is asked to select the time interval between the conditioned stimulus (CS) and the unconditioned stimulus (UCS), the number of pairings of the CS and UCS, and other factors.

These input parameters are used in the simulation to generate data, for example, the percent of subjects showing a conditioned response after each CS-UCS pairing.

Other possible tests may follow. The student may be asked if he wishes to test for extinction, spontaneous recovery, or related effects.

In general, the programs are characterized by a high degree of active involvement in the progress of the simulation, little necessity for pre-planning experiments (since the necessary information is supplied by the program), and ease of understanding and interpreting the results.

Because most of these simulations generate data for only one experimental condition at a time, the student who wishes to compare a treatment group to a control group or to a different treatment group must run the simulations twice: once for the treatment and once for the control. This limitation may actually be an advantage, since the student must have more than a passing familiarity with experimental methods if an experimental project is assigned. In addition, they can be effectively used as demonstrations for students who have not mastered the fundamentals of design.

The programs do not compute inferential statistics, nor do they keep records of students simulation activity. No gaming facility is provided.

LABSIM

The second set of simulations is connected to a program called LABSIM, written by Gerald Kissler and Richard Edwards of Washington State University. Originally written for batch use only, it was converted to interactive use at Albion last summer. This copyrighted FORTRAN program currently accesses 17 different modules or topical content areas.

Two separate programs are used in the system's operation. One of these programs creates the module file and student records file, and allows the instructor access to these files to obtain current records of students' simulation activities. This program is about 580 FORTRAN records. The other program is the one initiated by a student run. Our interactive version is about 2050 FORTRAN records. The original batch version we obtained from Rich Edwards is interfaced with the BMD08V analysis of variance program, and is thus substantially larger.

Initiation of this program causes it to request that the student specify which of the available modules the student wants. It then accesses the student records file and determines if there is sufficient money in the student's account to run the simulation. The gaming

features - including costs of running each subject and manipulating or controlling a variable - can be adjusted or bypassed entirely as the instructor specifies in the module file. If there is enough money remaining, the program goes into the module file and calls up the information contained there. This information consists of the list of the potential independent variables for the module and the effects associated with manipulating or controlling each possible value of the variable, as well as any interactions between them.

The student must specify a value for each variable for each condition in the design.

The program will generate data representing two-condition, one-way multi-condition, or two-way factorial designs of up to six groups.

Three types of variables are possible: discrete variables, such as treatment vs. no treatment, continuous variables, such as number of trials, or range variables, such as IQ, where the student must enter a lower and an upper limit of values for the variable. All three types may be left uncontrolled, in which case the program randomly selects one of its possible values and generates data based on this selection.

These data are computed using a linear regression equation, which also adds on a random error term so that all students' data are different from other students', even though the same design may have been requested.

Summary statistics (the mean and standard deviation) are printed, as well as a *t*-test for two-condition designs.

An evaluation of the design is given, which prints any design errors the program found, or if none were detected, the words "good experimental design."

Finally, charges to the student's account are listed. A "bonus" may be added to the balance remaining for good experimental designs.

In contrast to the stand-alone programs by Dave Johnson, no background information is provided at the terminal. In order to run these simulations the student must read a short handout describing this information. Also contained in the description is the scenario for the experiments and the list of potential variables available for manipulation and/or control.

Student Use and Evaluation

Approximately 200 students were enrolled in seven sections of Introductory Psychology at Albion College last term. Instructors assigned from two to five simulation reports in their sections. Based on the number of students per section and assignments per section, it was estimated that the minimum total of runs required to fulfill these assignments was about 750. The total number of simulations actually run was 1748, well over twice the minimum. Part of this difference can be accounted for by occasional failures to produce good designs on the first attempts. Nonetheless,

we believe these figures reflect the enthusiasm of our students about these programs, an enthusiasm which was also apparent in classroom discussions, questions, and other comments related to the programs.

The most popular of the stand-alone programs was the simulation of the effects of hypothalamic lesions on eating behavior in rats. It was run about 90 times. Across all of these programs available, the average number of runs per program was 41.

The average run time per program was a little under eight minutes, with a range of three minutes to 10 minutes.

Over 1375 simulations were run on the LABSIM modules then available. The overall success rate (percentage of "good experimental designs") was around 86%. The most frequently run of these was the module simulating experiments about "The Effectiveness of Study Activities on Retention," which allowed the student to manipulate variables affecting exam scores in a simulated classroom. It was run 287 times with a success rate of 77% (the lowest among any of those assigned). In one module all 45 of the 45 attempts yielded good designs. It was about programmed instruction.

The average connect time per student run of LABSIM was nine and a half minutes. Given that students averaged slightly over two experiments per sitting, this represents an average time of 4.5 minutes per experiment.

All together, the connect time for these programs was only 147 hours, or about three quarters of an hour per student. Contrasted with the amount of time that would be necessary to conduct actual classroom experiments, the savings is immense.

Student evaluations of the programs are available only from two sections of the introductory course, and these are all evaluations of LABSIM. Limited as they are, these evaluations reflect a highly favorable attitude of students toward the simulations. The instructor of these two sections (the first author) assigned as a requirement five reports, and left the option for students to complete additional reports for extra credit. The five simulations assigned were "about the right number" for the course according to 82% of the students, although 41% felt that they were not discussed enough in class. Most students (82%) felt that the simulations made a significant contribution to the course.

End-of-term course evaluations also reflected favorable student attitudes toward the simulations. Statements such as "The computer simulations were excellent" or "really enjoyed the computer experiments" were common.

Summary

Computer simulations of experiments in psychology can provide the introductory student

not only with substantive information relevant to the content of the course, but also with experience 163 in designing tests of hypotheses of their own devising. The programs available at Albion College are specifically designed for the introductory course. They are easy to use and readily understood by even the least computer-sophisticated student. Based on the number of unassigned runs, and on written evaluations, student reaction to the programs has been highly favorable. We feel that these programs provide instructors with an effective method of integrating scientific method with substantive content at the introductory level.

UTILIZING THE COMPUTER TO TEACH AND STUDY CLINICAL PROBLEM SOLVING SKILLS

Doron Gil, Chris C. Wagner
and John F. Vinsonhaler
Institute for Research on
Teaching
Michigan State University

Investigations of the process of clinical problem solving behavior in reading diagnosis have been underway for the past three years in the Clinical Studies Project of the Institute for Research on Teaching at Michigan State University. Such investigations are facilitated through the utilization of computers in the following areas: 1) the computer-simulation studies of clinical problem solving behavior intended to test the "Inquiry Theory" (Vinsonhaler, Wagner and Elstein, 1977) of clinical problem solving behavior; 2) the clinical problem solving behavior of human clinicians as observed and studied while they interact with either manual-based or computer-based simulated cases of reading difficulties; 3) the application of the knowledge gained from these interactions to the training by computer of clinicians (teachers, student-teachers, reading specialists) on diagnostic skills; 4) the data analysis of the different studies by computer.

Introduction

Investigations of the process of clinical problem solving behavior in reading diagnosis have been underway for the past three years in the Clinical Studies Project of the Institute for Research on Teaching (IRT) at Michigan State University (Gil et al, 1979). The main purpose of these investigations is to better understand the process of effective reading diagnosis and remediation by reading specialist, learning disabilities personnel and classroom teachers. This understanding may then be applied to the training of teachers and reading clinicians in diagnosis and remediation skills.

The Clinical Studies Project of the IRT employs computers in almost every aspect of its complex organization, from the more common uses (i.e., editing manuscripts) to more complex ones, such as computer-simulation of clinical problem solving behavior and the development and application of statistical analysis procedures for unique experimental situations. Such intensive use establishes the computer not merely as a helpful tool or aid in the execution of some technical processes, but rather as an essential ingredient of the project which makes it possible for such a complex organization to exist and function. In what follows we will attempt to describe the unique place that the computer has taken in our project.

A Theory of Clinical Problem Solving Behavior and Case Simulation

To start with, it is necessary to briefly describe the theoretical basis under which the project operates. The main thrust of the project is based on a theory of Clinical Problem Solving, the "Inquiry Theory" (Vinsonhaler, Wagner and Elstein, 1977). The Inquiry Theory, which first originated in medical education research conducted at Michigan State University (Elstein, Shulman and Sprafka, 1978), describes the behavioral domain in which a clinician (e.g., a physician, a reading specialist, a teacher) interacts with a case (e.g., a patient, a student) in order to reach diagnostic and treatment decisions about the case's problems. By describing the behavioral domain the theory attempts to predict those significant characteristics of the clinical interaction which will re-occur. The Inquiry Theory postulates that these significant characteristics are determined by the interaction of clinical memory (consisting of sets of problems, cues, treatments and the relationships among them), clinical strategy (the sequencing of the mental tasks performed by the clinician) and a case. A case is a patient, client, student, which exhibits certain behaviors.

A principle of this theory is that cases can be effectively simulated by using a set of data or information to represent a live case. Such a simulated case is based on real client and represent the relevant behaviors of the client's problems. The cognitive elements of such a case are: 1) a set of problems, 2) a set of cue names or items of information which

might be used to help a clinician make a diagnosis, 3) a set of cue values specifying the client's state with respect to the cue, and 4) a set of responses to all potential treatments.

The use of these simulated cases is a prerequisite for certain types of studies of clinical problem solving behavior. They provide the means for presenting the child's behaviors to the clinicians: different clinicians can interact with exactly the same case and their behaviors can be objectively measured and compared.

Since simulated cases are so important for the study of clinical problem solving behavior of clinicians, they should be constructed and put together in as much precision as possible. Also, on a large research scale there is the need for having several such cases, so that different studies with different cases can be run simultaneously. This is important in order to arrive at conclusions about clinicians' problem solving behavior which are not case specific.

Two forms of such cases are developed: manual based and computer based. The manual based case consists of various types of information about a child's reading performance stored in a box and manually retrieved. The information consists of background data, school records, different tests that the child performed, etc. As a potentially more cost effective form this manual simulation has been implemented on mini computer. With the computer based case all the information is stored in a computer file and can be retrieved via computer terminal.

In the process of creating such simulated cases, the information is collected by a reading specialist who interacts with a child on an individual basis. The reading specialist tests the child on various reading tests, and also collects some background information as well about the child. The computer then plays a major role in converting all this information to a simulated case.

First, the enormous amount of information is typed into a computer file and edited, so as to ensure that all the data is correct. Then, book indexes are run to check whether all the necessary information is included in the case. Finally, the data is printed, so that each piece of information is on a separate page. This process is necessary in order to have a systematic way of presenting the information to the clinicians (all pages being identical in form), and to enable easier storage in the case box.

With the numerous simulated cases necessary for our studies, it would have taken enormous time and man power to put together one such simulated case without the facilities provided by the computer.

Computer Simulation Studies of Clinical Problem Solving Behavior

Another area in which the computer is utilized in our project is through computer simulation studies of clinical problem solving behavior of clinicians. This is done by programming a computer to simulate a clinician and his interaction with simulated cases of reading difficulties.

Basically, computer simulation studies intend to test the "Inquiry Theory" of clinical problem solving behavior. According to the Inquiry Theory, the behavior of a clinician is a function of his memory and strategy (Vinsonhaler, Wagner and Elstein, 1977). Therefore, a simulated clinician can be built by simulating a memory and a strategy and their interaction, where changes in memory, in strategy or in both will be reflected in changes in the simulated clinician's performance. Computer simulation studies can therefore simulate a clinician as he interacts with a simulated case of a child having reading difficulties. Throughout this simulated interaction, the computer uses its memory and strategy to arrive at a diagnosis for a child.

In so doing, it uses a process of diagnosis similar to the one applied by human clinicians. The interaction between the computer clinician and a case starts when the computer receives the initial contact about the case (some basic information regarding his status in reading). With this information the computer proceeds to collect more information, on the basis of which it generates some hypotheses about the child's reading problems. Then the simulated clinician checks out these hypotheses using hypothesis directed inquiry (the process of collecting information to confirm or disconfirm these hypotheses). The simulated clinician continues collecting information and generating and checking out hypotheses until it reaches its final diagnosis.

With the simulated clinicians, simulation studies are run, in which different simulated clinicians may interact with various cases to reach diagnostic decisions. The diagnostic process and outcomes are analyzed, looking for implications of the known difference in the simulated clinicians and the observed difference in their performance. Consequently, our increased understanding of the clinical diagnostic process can then be applied to the training of human clinicians.

Two computer simulation studies were conducted so far by the Clinical Studies Project of the IRT (Vinsonhaler, Wagner and Gil, 1979). One study has tried to look at the impact of routine cue collection (the collection of some basic information about a case prior to the

166 beginning of hypothesis generation) on the clinician's performance.

Observational Studies with human clinicians have indicated that some clinicians use a routine cue collection procedure (or heuristic) at the beginning of every interaction with a new case; other clinicians do not use such a routine. The question then becomes, how significant is routine cue collection to the diagnosis process? To answer this question, several simulated clinicians were created and instructed to diagnose cases of reading disability using a routine cue collection. Others were instructed to diagnose the same cases but not to use such a heuristic. The performances of these groups of simulated clinicians as well as of human clinicians (who have diagnosed the same cases in one of the observational studies) were analyzed.

Results with respect to the accuracy of diagnosis indicated that simulated clinicians with routine cue collection significantly outscored those without such a routine, but still fell short of the human beings' performances. To the extent that the simulations used reflect an adequate understanding of the clinical problem solving process, the study suggests that reading specialists and classroom teachers may improve their performance if they use a routine cue collection as part of their diagnostic processes.

Another study using simulated clinicians investigated the processes of hypothesis generation as it is affected by different certainty thresholds for considering a hypothesis. Hypothesis generation is the process of selecting which hypothesis to consider in the diagnostic workup. The certainty threshold is how likely a hypothesis must be initially to be seriously considered during the diagnostic workup. The higher the threshold, the fewer hypotheses will be considered, because more and more hypotheses will have likelihoods which fall underneath this threshold. The lower the threshold, the greater the number of hypotheses that will be considered, because more hypotheses will have likelihoods above this threshold. This differentiation between high and low thresholds can be seen when in medicine, for example, students are trained to accept many hypotheses at the beginning of their encounter with a patient, and then narrow these hypotheses down to the most probable ones and study these ones very carefully. It means that these students will be more receptive to new hypotheses early in the session and less receptive as the session proceeds.

To capture this phenomenon, several simulated clinicians were created. Some of them, after receiving the initial contact and collecting the routine cues, were programmed to be more receptive to new

hypotheses in the beginning of the diagnostic session and less receptive to new hypotheses in the later part of the session. As the clinical interaction proceeded, these simulated clinicians were programmed so that their threshold of hypotheses acceptance rose. In the beginning the threshold was very low and many new hypotheses could be generated, at the end this threshold was very high which means that many new hypotheses could be generated.

By contrast, some simulated clinicians were created to be equally receptive to new hypotheses throughout the entire interaction. Like the previous clinicians, they started the interaction by receiving the initial contact and collecting the routine cues. However, when they generated hypotheses, the threshold of acceptance was low throughout the entire session.

The results in terms of time of hypothesis generation indicated that both kinds of simulated clinicians generated most of their hypotheses at the beginning of the interaction. This was an unexpected result. Upon closer examination, the process that apparently accounted for this behavior was the selection without replacement from a finite set of hypotheses (as the session progressed, fewer and fewer hypotheses that would relate to the cues existed). In other words, the rate of hypothesis generation had to decrease as the session continued. This simulation study would suggest that early hypothesis generation may be a necessary result of hypothesis-directed-inquiry.

These studies exemplify the type of information to be gained by computer simulation studies. They suggest that the use of routine cue collection increases diagnostic effectiveness, and that early hypotheses generation may be a result of hypothesis directed inquiry. These behaviors may be important to an effective diagnostic process, and both should be considered in the training of physicians, teachers and other clinicians on diagnostic skills.

Data analysis

A third area in which the computer is utilized is with the analysis of data from some of our studies, mainly the observational studies. In order to fully understand the importance of the computer in this area, it is necessary to first describe this form of studies.

Observational studies are an important mode of investigation through which the clinical problem solving behavior of human clinicians can be explored. This is done by introducing a clinician to a simulated case of reading difficulties and observing the interaction.

In a typical observational study a clinician is being asked to interact with a simulated case and arrive at diagnostic judgments about the child's problems. Procedures of such an observational study are typically as follows (Gil, Vinsonhaler and Wagner, 1978):

1) an observational session in which the clinician interacts with a simulated case of reading difficulties. He is given an inventory of information available for the case and is instructed to ask for any information he wishes that is listed in the inventory. As he examines the information he is encouraged to talk aloud and verbalize his thoughts about the case.

2) Following the observational session the clinician is asked to write down his diagnostic judgments about the case and a general plan for remediation.

3) The write up of the diagnosis is followed by a debriefing session, during which techniques of stimulus recall are used to recreate the clinician's problem solving behavior as he went along requesting and evaluating information about the case.

Data analysis of such observational studies can be divided into two: product measures and process measures (Wagner et al, 1979). Product measures are the final outcomes of the observational sessions, such as final diagnoses reached by the clinicians, hypotheses generated and cues requested. Process measures deal with what can be learned on the dynamics of the clinical interaction. These measures involve time dependent factors such as time of cue request, order of cues requested, average number of cues per session, and so on.

The main features of these analyses, in sequential order, are as follows:

1) Creation of data dictionary - the domain of all diagnoses, cues or hypotheses is stored on an MTS file;

2) Creation of data element vector - the transformation of the domain file into a numerical file that can be more easily processed for statistical analysis;

3) Calculations of the needed statistics (e.g., diagnostic agreement - how much do clinicians agree with each other on their final diagnosis for a given case).

These product and process analyses on such a large amount of data (several observational studies, each of which to be analyzed with respect to product and process measures as described above) can hardly be envisioned without the utilization of computers in the creation of the files and the execution of the statistical analyses.

Implications for the training of clinicians on diagnostic skills

Our experience so far with the observa-

tional studies indicates that clinicians' interactions with simulated cases may be valuable training experiences for them. However, with regard to training, there may be some disadvantages to the manual based simulated cases. Each simulated case is very comprehensive, and, therefore should be stored in two or three boxes. Whenever someone needs to be trained with a manual based case, these boxes must be checked for the presence of all materials in the proper place. Assuming that one wants to train a class of students with manual based cases, there is a need to have enough cases (or several replications of same case) to enable usage by all students. This raises the problem of reliable duplication and control over a large set of material.

Advantages of computer based simulated case are that the information composing a simulated case can be stored in several computer files (and hence we will have access to several identical simulated cases). Also, this information can be easily retrieved by instructing the computer to locate the needed information, and it will not be misplaced (because it is never actually taken out of its file). However, some disadvantages of computer based simulated cases for training purposes may also be noticed:

1) if the clinician who interacts with the case does not have the necessary training to handle a computer terminal he may take a long time to type in the appropriate key word in order to retrieve the needed information.

serve as a better visual illustration of a case than a computer based simulated case. It may well be that to the novice clinician who had never interacted with a simulated case nor has been trained with one it may be difficult to conceive the notion of a simulated case which presumably exemplifies a real patient. With a manual based case the clinician can actually see how the information is arranged in the box and perceive its resemblance to the real child.

However, when these problems are solved, the training by computer of clinicians seems viable. One advantage of the computer based simulated case is that it can be designed so as to provide the clinician with immediate feedback as he diagnoses a case, i.e., directing him toward hypotheses he should consider about the cases' problems, toward cues to collect, etc.

One exploratory training study was conducted by the Clinical Studies Component of the IRT during the summer of 1977 (Sherman, 1978). The major objective of this study was to investigate the impact of simulated cases and decision aids on the diagnostic performance of prospective teachers. In addition to the traditional presentation of course content (in a

reading diagnosis course) the students interacted with manual based and computer based simulated cases of reading difficulties. Results indicated that the students did learn to diagnose in a manner similar to reading specialists, as indicated by increases in their mean diagnostic score. It was concluded that manual based and computer based simulated cases can be used in the training of clinicians on diagnostic skills.

Summary

This paper has described a few areas in which computers are utilized as a major ingredient of a large research program. It attempts to show that the role of computers should not be viewed as merely an additive one to the studies or training which are carried on, but rather computers may play a more fundamental and central part. Not only can they assist us in the preparation of our research tools, but they can also be the mean for providing us with better understanding of the clinical problem solving behavior of human clinicians (i.e., via computer simulations). Also, computers seem to have the potential for becoming a major source in the training of clinicians on diagnostic skills.

References

- Elstein, A.S., Shulman, L.S. and Sprafka, S. Medical problem solving: An analysis of clinical reasoning. Cambridge, Harvard University Press, 1978.
- Gil, D., Hoffmeyer, E.C.M., VanRoekel, J.L. and Weinshank, A.B. Clinical problem solving in reading: Theory and research. East Lansing, Michigan, Institute for Research on Teaching, 1979.
- Gil, D., Vinsonhaler, J.F. and Wagner, C.C. Studies of clinical problem solving behavior in reading diagnosis. Paper presented at the First Annual Behavioral Studies Conference, University of Missouri- St. Louis, October 20-21, 1978.
- Sherman, G.B. Training reading specialists in diagnosis. Paper presented at the Annual Meeting of the American Educational Research Association, Toronto, Canada, April 1978.
- Vinsonhaler, J.F., Wagner, C.C. and Elstein, A.S. The Inquiry Theory: An information processing approach to clinical problem solving research and application. Medinfo '77, (Proceedings of the Second World Conference in Medical Information), 1977, 157-160.
- Vinsonhaler, J.F., Wagner, C.C. and Gil, D. Studies of an information processing theory of clinical reasoning via computer simulation. Paper presented at a conference on "Evaluation of efficacy of medical action", Bordeaux, France, May 14-19, 1979.
- Wagner, C.C., Kitchen, F., Polin, R., Vinsonhaler, J.F., VanRoekel, J.L. and Gil, D. Observational studies: Data analysis system. Technical Report.

East Lansing, Michigan, Institute for Research on Teaching, 1979.

"WE DELIVER" (JOBS)

Robert J. Miller
Gerri Powell-Rae
Washtenaw Area Youth Placement Service

Roland L. Meade
Washtenaw Intermediate School District

The Washtenaw Area Youth Placement Service (WAYPS) was established in October 1977. A main function is the coordination of job information and delivery to local school placement offices via the Washtenaw Intermediate School District (WISD) data processing facilities. Primary users of the "Job File" are placement and co-op coordinators in each high school, and vocational teachers and students. Michigan Occupational Information System (MOIS) numbers are used for job clustering and a tie-in with the MOIS program. The nature of Area Placement funding necessitated quick implementation of the program. WISD support was critical in utilization of computer resources for the program. Simultaneously, local schools purchased terminals for MOIS and Placement with the help of CETA funding. The significance of the project is the cooperation of WISD, CETA, MOIS and WAYPS to provide current job information on a county-wide basis.

Introduction

In the last few years the high unemployment rate of youth has focused increased attention on the need of youth for job placement assistance. As the majority of youth (70% in Michigan) enter employment during or immediately after high school, the strategic time for placement assistance is at the high school level.

A major effort to increase the number of students to whom job placement services would be available was the provision of funds by the Vocational Technical Education Service (Michigan Department of Education) for Area Placement Projects in 1975. The Area Placement Program was intended to be a system which coordinated and supported job placement activities of schools within a natural labor market area.

A few of the very successful Area Placement Projects "pioneered" the use of the computer with data files of job openings, employers and graduates. With few exceptions the computer storage and retrieval systems were operated out of Area Vocational Skill Centers and served the vocational graduates at those centers.

In October 1977, local schools in Washtenaw County received funding for the

Washtenaw Area Youth Placement Service (WAYPS). The funded proposal included use of the computer for job placement activity. As Washtenaw County has no Area Vocational Skill Center with a centralized placement office and body of students, the area placement function is that of helping local schools and their placement offices.

A key element of the Area Program is the coordination of job opening information and delivery to local school placement offices. For the first months of the Project, job opening information was typed, copied and sent by the ISD mail system to the local schools. In order to speed-up the process of delivering job information it was decided to proceed with the development of a Job File as first priority.

The possibilities of a student file and employer file had been given serious attention for months but the delivery of job opening information seemed more consistent with the assistance and coordination being provided to the local placement offices. With the priority of the Job File determined, the Washtenaw Intermediate School District's (WISD) data processing staff, together with the Area Placement Office began the process of deciding what data was needed and what system to use.

It was decided that the FINDIT system would best suit the needs of storing and maintaining WAYPS job information. FINDIT is part of the Hewlett-Packard contributed library. It runs on the Hewlett-Packard 2000 series computers. However, this file could be maintained by any file handling system. If a general purpose file-handling system is available at a site, a minimum amount of programming would be required to duplicate the system described in this paper. An added advantage to this approach is that the programmer can tailor the program to meet the needs of any given group.

FINDIT allows the file manager to add records, delete records and update records within the file. The record format can vary. The following list is the format used by WAYPS:

- 1 ID--Job number
- 2 FLAG--Status (open or filled)
- 3 DATE
- 4 TYPE--Full or part-time,temp,summer
- 5 JOB TITLE
- 6 DOT--Dictionary of Occup. Titles Number
- 7 JOB DESCRIPTION
- 8 CONT--Job Description continued
- 9 HOURS
- 10 SALARY
- 11 EMPLOYER
- 12 ADDRESS
- 13 CITY
- 14 STATE
- 15 ZIP
- 16 CONTACT
- 17 PHONE
- 18 EMPLOYER TYPE--USOE cluster
- 19 DATE FILLED
- 20 MOIS NUMBER
- 21 SOURCE--Where job originated

This was not the original list of elements that was recorded for every job. Since the WAYPS staff was not familiar with computer programs, much less distribution of jobs via computer, the beginning stages were trial and error. The above list is the result. Soon there will be a twenty-second element called "number placed". This will alleviate the problem of keeping track of multiple job requests, e.g., "Waitresses/Waiters -- 3 needed". FINDIT does not limit the number of elements, but all of the data for one job must fit in one record. On the 2000F the record size allows storing between 400 and 500 characters of data. This has been adequate. FINDIT allows multi-level searches on the data. For example, one can request a list of all jobs that have been posted since a given date.

But there are a limited number of ways for the information in the file to be printed

out. Every time a placement or co-op coordinator accessed the "Job File" it printed out in the "list" form (as shown in the list of elements), with each piece of information appearing on a separate line. It was not only a waste of paper and time (generally there are between 80 and 100 jobs listed) but also it was too bulky to be used. Since the format of the listings available from FINDIT was not practical, the placement and co-op coordinators who used the system were asked for suggestions. A part-time programmer was hired to develop programs to access the data base and print formatted listings of the jobs. These programs access the Job File and prepare an index file. The final product is either a coordinator or student list.

The coordinator version prints in reverse chronological order. Beginning with the first job entered, the user can quickly note new additions to the job list. An "update" version is also available, whereby the user can receive the latest additions and deletions of jobs. The updated listing eliminates the need to run off a complete job listing every time the Job File is accessed. (A complete coordinator version can take 20 minutes to list.)

The student version of the job list differs from the coordinator version in several ways. This list does not contain all the information necessary to make contact with the employer. It provides the student with the job title, description, salary, and location--if the student is interested in the position, s/he must see the coordinator for complete details and/or a referral. This list is printed by vocational cluster. Each cluster appears on a separate page and begins with the vocational heading e.g., Auto Mechanics or Business Office/Bookkeeping. Clustering and suppressed employer information makes this list ideal for posting. Coordinators find it useful in promotion of placement services and vocational teachers find it helps bring the world of work into the classroom.

The vocational clusters of the student list were determined by MOIS number (element 20). Each job is given a MOIS number which corresponds with one of 350 occupations described in the Michigan Occupational Information System. The MOIS system is available either on computer or microfiche. If the student sees a job s/he is interested in while looking through the Job File, s/he can access the MOIS program and use the MOIS number to search on the occupational requirements, duties, working conditions, employment outlook and possibility for advancement for that particular job. Then with that added

information, the student can decide if the position is worth pursuing. It will also help him/her prepare for an employment interview and for making an evaluation of a job offer.

Generally, coordinators were hesitant about using computer terminals, at first. But after careful explanation and in-servicing, they seem comfortable with the delivery system. Students have no qualms about using the terminals; the only problem area here is keeping the access code for the Job File confidential. Samples of the printouts are listed at the end of this article.

Terminals in Local Schools

The question is frequently asked about the availability of computer terminals in local schools. For years the Ann Arbor Public School System have had terminals for both instructional and college search (CIVIS) purposes. Other school districts had terminals for instructional purposes and possibly these could have been used for the Job File.

Two developments at approximately the same time prevented the problem of not having a terminal available. MOIS was available in the computer version through the WISD time sharing system. Considerable interest and enthusiasm had been aroused in the MOIS computer version and the Ann Arbor High Schools were already using the system.

The other development was the federal youth legislation which included the Youth Employment Training Program. The Washtenaw County CETA allocated the bulk of its YETP funds for in-school youth. In addition to the work experience, emphasis was given to transitional areas of youth employment preparation which included placement, employability skill training, vocational/career counseling and occupational information. With costs shared between schools and CETA, many schools purchased terminals for exclusive use with MOIS and the Area Placement Job File.

Job File System

The rationale for a centralized Job File is to provide employers a central point for posting job opening information and to provide a system by which placement coordinators can share unfilled job openings with the other high school placement offices.

The system starts with job openings called

into the area office. The phone calls may be from area employers or from the local school placement offices. The necessary information is recorded on a "job order" card. Then daily or as soon as possible, the job information is entered into the WISD computer and is immediately available via terminals in the local high schools. 171

When the job information comes from an employer a post card is sent acknowledging the job order, thanking the employer and notifying the employer of his/her responsibility.

The area office regularly calls employers who have listed job openings to determine the job's status. That information is coded and keyed into the Job File:

- 0--OPEN
- 1--FILLED BY PLACEMENT SERVICE
- 2--FILLED BY OTHER
- 3--WITHDRAWN BY EMPLOYER
- 4--WITHDRAWN BY PLACEMENT SERVICE

The system does not provide for absolute control of the number of referrals to an employer. The philosophy of WAYPS is to support the local placement programs and to encourage communication directly between employer and placement coordinator. The coordinator is therefore strongly encouraged to call an employer listed on the Job File before referring youth.

Such contact:

- opens communication channels between employer and coordinator
- verifies job opening status
- provides opportunity to elicit additional information on job opening
- provides opportunity to give employer information on the job applicant
- provides opportunity to determine number of referrals desired

Assuming that school staff has always cared about students and tried to help them with their employment needs, it's obvious that channels of communication with neighboring schools (unless in the same school district) were seldom used and coordination of effort rarely occurred.

The WAYPS Job File is a means of sharing information which can benefit youth with a greater number and wider selection of job openings. It also benefits employers with coordinated access to more students which ultimately benefits students and schools.

However, the system is dependent upon the

172 sharing and cooperation of placement and co-op coordinators. Through monthly in-service meetings coordinators meet to improve their skills, share information and ideas, and build mutual trust. As coordinators learn to respect and trust other school coordinators the sharing of "quality" job openings and employers benefits all students.

Results

The data is inconclusive but after one year of operation there are a number of positive outcomes. Six-hundred and sixty-three (663) job openings were entered into the Job File in the first year, ending October 1, 1978. Twenty-four percent (24%) of the openings or 157 were identified by local placement staff and were redirected to the other local offices via the Job File.

Twenty-four percent is somewhat impressive and reflects a beginning of coordination and sharing. However, the impact is diminished when realizing that only 6% of all job openings identified during the

year were redirected. It is unlikely that 94% of all job openings were filled by local placement offices.

The bottom line of the program is placement. Twenty-five percent (25%) of all jobs listed in the Job File were filled by youth. No doubt a much higher placement rate could be achieved if the area office had direct access to students. However, the WAYPS philosophy is to strengthen and support the local placement office.

It is believed that eventually the local offices will provide placement services on a regular twelve-month basis and the Job File will be an integral part of the service.

Determining the effectiveness of the system has been a continual concern. At the present time, coordinators are keeping records on which job openings in the Job File they refer youth to. Feedback will help the area office develop jobs relevant to the employment needs of youth.

COORDINATOR

Would you like the full listing, the updated listing, or both--full
Set your paper to the top of the next page and press return

>>> THIS LISTING CURRENT AS OF 03/08/79 <<<

<<<< INDEXED BY JOB # >>>>

***** JOB #709 POSTED ON 03/07/79 *****

CLERK (PT-TEMP) (MOIS #46)

WORK WITH MAILING LABELS (JOB LASTS 4-6 WEEKS)

SALARY : \$3.29/HR

HOURS : HALF-DAY (AM OR PM)

CITY : ANN ARBOR

Contact:

NEVA JORDAN

GREAT LAKES FEDERAL SAVINGS

401 E LIBERTY

ANN ARBOR, MI 48104

Phone : 769-8300 EXT 258 OR 259

STUDENT

<<<< AUTOMOTIVE/MECHANICS >>>>

***** JOB #336 POSTED ON 07/14/78 *****

APPLIANCE REPAIRER (FT) (MOIS #160)

LARGE APPLIANCE REPAIR, MICROWAVES, WASHERS,

DRYERS, STOVES, WILL TRAIN MECHANICALLY INCLINED PERSON

SALARY : \$2.65/HR

HOURS : 8-5, MON-FRI

CITY : ANN ARBOR

***** JOB #563 POSTED ON 10/06/78 *****

AUTO SERVICE TECHNICIAN (FT) (MOIS #291)

CHANGE TIRES, BATTERIES, SCHOCKS, ETC

SALARY : -

HOURS : FLEXIBLE

CITY : ANN ARBOR

COMPUTER - RELATED CURRICULUM

SUMMARY OF PRESENTATION*:
THE ASSESSMENT OF STUDENTS' COMPUTER LITERACY -- RESULTS
FROM THE MECC COMPUTER LITERACY STUDY

RESEARCH GROUP MEMBERS:

Ronald E. Anderson, University of Minnesota
Thomas P. Hansen, Minnesota Educational Computing Consortium
David C. Johnson, University of London
Daniel L. Klassen, Minnesota Educational Computing Consortium

ABSTRACT: As part of a Computer Literacy Project being conducted by MECC, a comprehensive computer literacy test has been developed by the project team and administered to over 1400 Minnesota secondary students. Pre- and post-test results associated with various approaches to teaching with and about computers have been analyzed. The test deals with attitudes towards computers and with the cognitive areas of hardware, software and data processing; the application of computers in society; the impact of computers upon society; and algorithms and programming. The two-year Computer Literacy Project is supported by the National Science Foundation.

BACKGROUND OF THE STUDY: The Special Projects Division of MECC is nearing completion of a two-year Computer Literacy Research Project that is supported by the National Science Foundation's Research in Science Education program. The Project team is investigating the impact of pre-college educational programs that are designed to increase computer literacy and the effects of human-computer interaction within the context of instructional computing environments.

COMPUTER LITERACY OBJECTIVES: Over the past year and a half the team has developed a set of sixty Computer Literacy Objectives for the secondary school level. As a first step in developing the objectives, information on a wide range of courses was collected (United States and Great Britain) through personal contacts, responses to requests for information which appeared in various journals, and communications with the institutions cited in the "HUMRRO Exemplary Instructions" book (Human Resources Research Organization, 1977). In addition to course descriptions, nearly 2,000 test items relating to computers were collected, judged for quality, and categorized. The course descriptions and test items were then used to develop a comprehensive list of the most prevalent topics and objectives covered in the courses. These were grouped under six main headings: Hardware; Programming and Algorithms; Software and Data Processing; Applications; Impact; and Attitudes, Values, and Motivation. The first five headings deal with cognitive outcomes and the sixth is concerned with attitudes or the affective domain. The list of objectives was subjected to many iterations and then circulated to 12 "outside experts" representing the professional societies, industry, and education for reactions and validation. On the basis of feedback from these individuals, the list was revised once more and the current version developed.

MINNESOTA COMPUTER LITERACY AND AWARENESS TEST: Based upon the objectives, a 100 item test was developed to measure student computer literacy attitudes, knowledge, and skills. In September, 1978, the test was given as a pre-test to 1438 secondary students in Minnesota. These students were in sixty classes that represent four different approaches to the use of computers in classrooms. The approaches included:

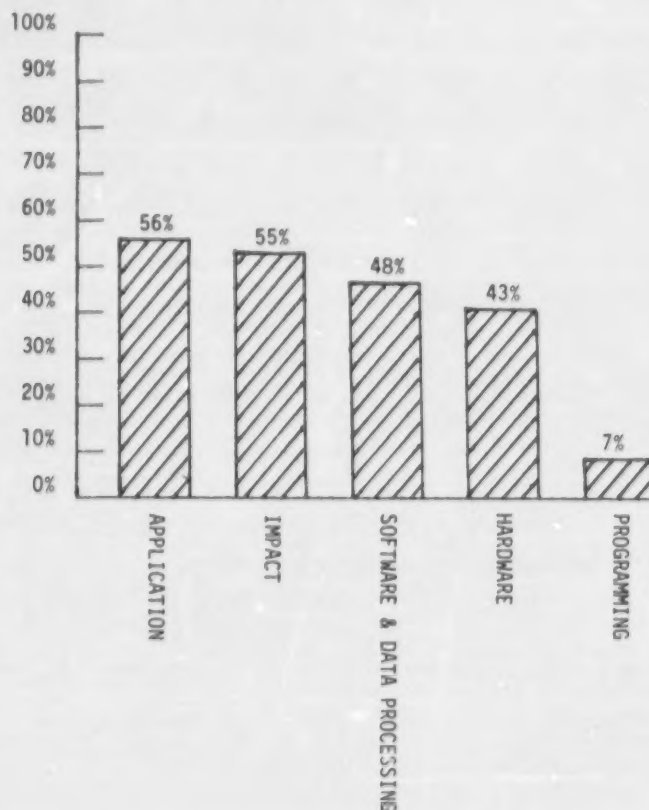
- (1) Use of the computer as a tool in the classroom.
- (2) The teaching of programming.
- (3) Teaching about the computer in society.
- (4) A comprehensive approach involving several components of the other three approaches.

By January all of the students completed the test again as a post-test.

PRE-TEST RESULTS: The initial analysis of the pre-test results indicates that students could answer fewer than 50 percent of the questions about computer hardware, software, programming, and algorithms; the application of computers in society; and the impact of computers upon society. For example, only 30 percent of the students knew that computers are used to commit crimes, especially stealing money and stealing or falsifying information. Also, only 22 percent knew that a person does not have to know

*A paper presenting a complete analysis of these data will be distributed at the conference.

176 how to program to use a computer. The students also had difficulty with the meaning of terms such as software and hardware. The students performed best in the areas of application and impact, and poorest in the programming section of the test. Twenty-seven percent of the students attempted to answer the programming items, but only a third of these answered the items correctly. Overall, the mean score for the 49 item cognitive portion of the test was just over 21 items correct. The figure below shows the average percent correct for the five groups of items.



POST-TEST RESULTS: An analysis of changes in student attitudes, skills, and knowledge related to computers has been made. The various approaches to computer use in the classrooms have been compared with regard to the specific pre-post changes that occurred as measured by the test. The analysis attempts to answer the questions such as whether a student's conception of the capabilities and limitations of computers is affected by learning to program in a computer science class, by running simulations in a science class, or by studying about computers in a social studies class. Further analysis of the affects of a multitude of variables such as students' past computer experiences, favorite subjects, and fear of computers will be made.

BREAKING DOWN THE BARRIERS THE REAL MICRO MIRACLE

Darlene E. Crawford

Independent School District 196
Rosemount, Minnesota

Abstract

As the miracle of the micro revolution envelops the educational community, it forces educators to make immediate decisions regarding the value and use of this marvelous computer. It is imperative that the micro be used to solve the real problems of education-BARRIERS - the barrier between home and school and the barrier between teacher and computerization. Hardware selection, curriculum impact, consulting, in-service training, lesson planning, piloting and evaluation are procedures adopted in implementing the plan. The outcome should foster an educational philosophy that includes parents in educational decision making and enables students to cope with a highly technical and computerized society in the year 2000, while assisting teachers in viewing the computer as a valuable tool that will free them to participate in human kinds of activity.

50 years ago the educational institution was perceived as the authority on education and parents backed the premise 100%. 25 years ago parents had some questions and were invited to participate to a small degree in their child's educational process but there was not really an active impact. In the last 10 years, the parental community has stood up and said it wanted to be counted as an integral part of their children's educational process. Citizens committees and parental groups have sprung up in all areas of the educational field. Buildings no longer appear based on administrative or school board perception alone; curriculum changes constantly include teachers and parent input; transportation of students has been impacted greatly by parents' involvement. What is happening to the school's authority and credibility? Community action has challenged them to further accountability in every facet of education. Parents are demanding that their children read, that they have equal opportunity to participate in athletics, and other extra-curricular activities; that the curriculum be expanded to include daily living practices, i.e. family living courses and developmental psychology courses. This sounds as though parents really have a say in their student's educational goals and objectives but the fact remains that even though we have made tremendous strides in the past few years, a real barrier remains between school and home. This barrier needs to be broken down drastically in the near future if we as educators are sincere in preparing the student for every aspect of life as it will be in the twenty-first century!

Van Mueller of University of Minnesota speaks of Futures in Education regarding school finance and one of his statements was "families will choose their school and teachers in year 2000". Some school districts allow their families to choose the school and the teachers now, so why wait for 2000 - let's start now and be ahead of the times. Howard Karlitz, in the October 1978 issue of Kappa Delta Pi Record, issues a challenge to parents to take an active role in the educational community. That role should not only dictate specific political behavior but behavior related to the process of education itself. He says "Do not surrender or delegate to the 'experts' one's educational decision-making authority as a parent." Hallelujah!

Before someone accuses me of heresy, let me say that no one values an education more than I do - but education can come in many ways. I remember when I sent my first child to school back in 1959. The teacher told me that I should have never taught him how to print because I had done it improperly and now he has to learn over. Ditto for numbers. Well, I wonder how a teacher may feel now when the student comes to kindergarten from his experience with something as simple as Sesame Street where he learns numbers and letters. What will the kindergarten teacher do when the child will learn how to read before he comes to school? What will the teacher do when the Little Professor or Dataman has already taught the 3 or 4 year old how to come up with the division, multiplication, addition, subtraction sets and Speak & Spell has taught them spelling. I suppose the teacher can feel useless - OR -

178 educators can recognize the inevitable and plan now for a new mode of instruction. We must stop trying to justify the system, spot the ills and cure them. Technology can aid immensely in this task. Computers should calculate, bear pressure, help in decision making, stamp out pieces, pack and do all those menial tasks that it can possibly accomplish so people can participate in what is enjoyable. Our nation is born of the work ethic - in order to survive - in order to be worth anything - in order to succeed - in order to get powerful - in order to get money - you must work, work, work. WONDERFUL - as long as you are working at what you like, what is beneficial to you and yours as people - not work horses. Computers should be used to accomplish that end. If the instructional system does not meet the needs of the twenty-first century, our students will be growing up with an outdated method of instruction to live in a modern world of technology that will not only overwhelm them, but perhaps control them. If they are going to be effective leaders in our future society, then we had better start now to show them the way - to equip them with the tools necessary to make decisions based on this reality.

Another heavy impact on education will be the changing family structure. 60% of the mothers in this nation will be members of the work force by 1985. The percentage of single parents is on the increase. I remember when our Census clerk in our Office could count on a couple of hands the number of students with last names different than the family. Now it takes cross reference lists and a separate file to track this kind of student. This is another positive fact that the family structure is changing. These children will always come to school but now they'll need more personal care and babysitting than ever before...and it looks like these needs will far outrank their academic needs if the trend continues. Motivation, back patting, problem solving (the personal kind - not the math kind) will be teacher functions. The days are gone when we can insist that schools are not responsible for parenting. Parents may want to do it but find it impossible. The economic picture dictates that many mothers work today. Owning a home for the young people in the near future means both work and maybe two jobs.

Another barrier that needs breaking is the barrier between the instructor and the computer. The perceived threat by teachers as computers are suggested to be instructional tools is a reality. They see it as a replacement - a loss of job and security - a taking over by the in-human element. Education in this area has been minimal. We need to overcome this fear by showing the instructor that the computer is no more a threat than the slide projector. The computer is the greatest technological advance in years -

particularly - the micro computer. It can release the teacher from oppressing tasks so that time can be used to further humanize education. There has been such bad utilization of computerization that some teachers have had poor experiences - therefore their perspective is cloudy. Teachers use to mark report cards manually in the teacher's lounge or had individual contact with each student as they graded them. Then along came the new time saving, up-to-date mark reporting system! It was to save them time and effort. Unfortunately, instead of time-saving and efficiency, the teacher found a cumbersome, difficult to understand system that often times provided incorrect information. Of course, the standard answer came, "the computer made an error". Personal experience shows that the computer did what it was told to do by some programmer who wrote a mark reporting program based on what he decided the needs were. Then a service bureau sold the system to a school district superintendent who enthused his principals by stating some record keeping advantages that may or may not be true. To the teacher there is no real advantage, the work load is new and increased and suspicions relative to computers and their task capabilities grew. It is this mindset that needs altering if we are to help teachers fully realize the unending potential these micros have to assist them, relieve them, teach them, BUT NOT TO REPLACE THEM! They need to learn that teachers are the masters, they tell the computer what to do, when to do it, how to do it and that they have the power to shut them down when all else fails.

As I stated previously, the educational community went unchallenged for years thereby allowing a prevalence of complacency in all areas of administration. These attitudes sponsored the causes of the barrier we have today - academic minded instructors and administrators; untrained or academically oriented. There were no managers, decision makers, trained executives that looked on the educational world as a big business; ran it as such and felt accountable to the taxpayers for the way in which they spent the dollars. There is some advancement in this area, but it is minimal. This kind of administration fosters barriers between home and school.

Increased levels of union activity on the part of organized teaching and administrative personnel is another large cause of barriers between home and school. Parents who are being asked for more money with less accountability and no active part in the decision-making do not see walls coming down, but getting higher. They feel frustrated by a situation that seems to have mushroomed into unhappy personnel strikes, shorter hours by teachers, and less and less cooperation from administration and instructors when they call the school.

Decreased levels of state and federal aid to school districts has created a monumental problem for administrators, school boards and staff as they face decreasing enrollment and school closings, in addition to cut-backs in many levels of support. The mis-management of funds by incompetent administrators and/or school boards fostered much of the cut-backs we are now experiencing as a result of untrained personnel.

Taxpayer revolts have pressured educators regarding more accountability. Taxpayers are demanding more for their money and the money is getting harder to come by.

The authoritarian attitude of administrators has propagated a decision making atmosphere that excludes not only parents and students but also instructors and often times board members from any participation. This kind of planning does not assist the instructor in dealing with new and innovative ideas in curriculum let alone prepare them for the impact of the computer on their teaching experience.

The appearance on the scene of the micro computer seems to offer some beginnings to a solution to the problems stated above. It is available to school and home, to experienced user and novice, to industry and education. It comes in attractive outfits, is comparatively inexpensive, and poses less threat than ever before. It is wonderful, marvelous, challenging, frightening, overwhelming but it is not a fad. It is here to stay and if used properly, it can turn the educational community around in its approach to students, parents and teachers.

If handled properly, the micro has the potential to do all the following: evaluation and monitoring of student and staff, record keeping tasks too numerous to mention, lesson planning, menu planning, assist with energy saving, store and find information, provide data to assist in decision making, act as a catalyst to bring the teacher closer to the student and home, serve as a communication device between school and home, offers excellent video capabilities complete with color and graphics, adds a sound dimension enabling musical outpouring, and heading toward voice recognition possibilities already in use but not refined.

Ultimately, all the tasks performed by the micro should free a teacher to involve himself with students and other teachers in a more personal and human way. Once the onus of menial tasks is reduced to a minimum, the instructor can then share with, listen to, and care about students in a personal way. It should free teachers and students to do what people should do - talk, listen, reach out, confide, grow, relax, smile and love. This will only come to fruition after extensive education and in-service training of the professional staff that their changing role can be met

and conquered but certainly not dispensed with. 179

We have some of the best educational facilities in Minnesota - particularly in Rosemount. We have come a long way in administering to the special needs of students, but a miracle has happened - the micro miracle - and it has the ability to so drastically change the whole lifestyle of this country that it truly is mind boggling. It is here to stay, it is not a fad, it is a sweeping technological advance such as our society has not seen since the splitting of the atom. What was suppose to be the space age really turned out to be the computer age - and more specifically the micro computer age. It will have its impact on society as computers never have before. Computers in the past have always been that big machine somewhere out there that does an inconceivable job of messing up my phone bill or bank statement. It was perceived as a threat in most instances - but it came and it stayed. The micro miracle changes that picture drastically - people are shopping in the personal computer shops - they are becoming less frightened - they see fun, games and real aids to running their lives in these marvelous little pieces of equipment. They are bringing them home. Many of those kids will come to school with knowledge of what a micro can do in the next five years. Will we be ready for them? Will teachers overcome their fear in time to view the computer as a valuable aid?

Technology will force the educational structures to change in this society in the near future. By 1985, says Bob Albrecht, 10 million personal computers will be placed in the homes; by 1990, says Earl Joseph anyone can carry a pocket computer around with an unbelievable wealth of information at their fingertips. Educators can be in on the ground floor and try to lead the revolution in learning processes or they can sit back and wait for technology to revolutionize the educational institution and then react to it. One thing is certain; the process by which students will learn is going to change. Picture a classroom where a group of students come to school to tell the teacher what they learned on their micro last night. Envision a teacher who then acts as the catalyst for this group of students to pass the information between them. I am sure this seems frightening to many people - parents as well as teachers - but I do believe it will become reality. Why should a teacher spend hours formulating lesson plans, lecturing, grading, testing and forcing academic learning in a structured classroom when a student can sit down at home, access a library of information and find out anything he ever wanted to know about anything by pressing a few commands on a micro - or even better yet - a hand held computer that has enough memory on a small chip to store a treasure of information.

The first cry that goes out from educators when you talk about computerization is "We can't afford it" and "Where do we get the money?" In our district, we did a couple of things - we started looking at how much money we were spending on other subject areas, i.e. - biology - do you have any idea how much money has been spent on microscopes, lab manuals and last, but not least, live frogs and chickens for experimental purposes? Check it out in your district. How many math curriculums and reading curriculums has your district tried and discarded in the past 10 or 15 years? Now justify that in accordance with the dollars spent on computer literacy and impact on society. Investigate, educate, prod your board of education, your administration, your community to take another look. Don't just jump to the conclusion that it's not affordable. Next we looked at how Title IV-B money is being spent. It is money federally granted and available to school districts throughout the nation. The administrator in charge of Federal Funds makes a proposal to your State Department of Education stating ways in which it proposes to spend the money. This is where we have received the majority of our funds for the TRS-80's and Commodore Pets.

In addition, we have been willing to designate consultant funds for specific assistance. We picked a consultant carefully. Teachers are not inspired by consultants that walk in and tell them that they have all the answers; nor are they inspired by a consultant that says to them "Write your own curriculum". We found a happy answer and a middle of the road attitude in the person of Norm Bell of Michigan State. He comes with expertise in the field of handling people problems, formulating ideas, putting them on paper and finally into action. He listens, writes, listens again, re-writes, pilots re-writes and constantly evaluates what kind of program is being offered. Teachers from all grade levels served on a computer committee to aid, assist and challenge Norm to produce what we feel are some soundly based lessons dealing with computer literacy and impact on society. It was not a simple task; it took months of planning, research, piloting, evaluating, and revamping before we introduced it as a part of the curriculum.

Scope and sequence is another area that needs addressing when a curriculum is impacted as heavily as this. Our pilot was conducted from first through seventh grade and was decided finally to be used in the fifth grade level. Immediately upon that decision, we began making plans for curricula that could be implemented at the first grade level all the way to twelfth grade. As the learning becomes more sophisticated the students become more aware at a younger level. As the students progress through grades it is imperative that they have opportunity to review but not repeat information.

It is extremely important to include teachers in placing curriculum at a grade level. They don't always agree but if they feel a part of the decision, the way in which they handle the lessons is certainly more positive. It is vital to remember that no decision is cast in concrete, so if a move needs to be made in the future, the opportunity to do so should always be left openended.

Our district is just beginning to meet the challenges of breaking down the barriers I have defined above! We are looking forward to the time when our final goal is met - students, parents, teachers and administrators all working together to learn from each other. Our philosophy of education will be finally to prepare our students to gain the knowledge necessary to run this country in the twenty-first century.

Before you leave, consider this. Times are changing - education is changing! Technology is changing our educational scene in such a rapid manner it is mind-boggling! Will you be reacting to or leading the revolution?

CAI LITERACY IN INSTRUCTIONAL TECHNOLOGY CURRICULUM

Marsha Young
Rochester Institute of Technology

Growing acceptance of computers as an instructional medium is demonstrated by the inclusion of a course on computer assisted instruction in the instructional technology (IT) curriculum at Rochester Institute of Technology.

Students are introduced to CAI terminology, sources of current literature on CAI, instructional applications, instructional strategies, and on campus hardware. Students receive demonstrations of CAI on large and small systems and demonstrations of commercial and locally produced courseware. Students are taught flowcharting, a computer language, inputting and editing skills. They then have the opportunity to apply instructional development skills learned in the core IT courses to a short CAI segment; code, input and debug it; use classmates for a trial student run; and write documentation and recommendations for revision.

This paper describes the course and discusses the rationale of its content.

Students of instructional technology can gain insight into the uses of computers for instruction by authoring a small CAI program as part of a course in computer assisted instruction. Using a systematic instructional development model, the students synthesize and apply skills learned in previous IT courses: objective writing, task analysis, criterion testing, content analysis, formative evaluation, while gaining perspective of a new and evermore available instructional medium.

The instructional technology program requires students to take a core of courses in instructional development theory plus elective courses in learning style, psychology of learning, interpersonal communication, research, instructional facility design, management and budgeting. Proficiency in media production skills such as 35mm photography, Super 8 film photography, audio and video recording, graphics and overhead transparency production are assumed as entering skills but may be obtained through supplementary courses and practicum. Courses in programmed instruction, instructional television and computer assisted instruction are advanced standing electives. Programmed instruction is encouraged as prerequisite to CAI to develop an awareness of pacing, small step instruction and mastery learning. The major focus of the department's graduate curriculum is to prepare individuals for successful employment as instructional materials specialists or instructional development specialists and to extend and update the ID skills of professionals already working in the areas of training and instruction.

What benefits would such students gain from a CAI course?

- Exposure to a medium that is becoming available

in more and more training and academic environments and in homes.

- Objective investigation of the history and application of this medium without the pressure of bias advocates.
- A sensitizing to the potential and characteristics of the medium.
- An awareness of the variety of systems and software available.
- An awareness of the resources (skill, time & money) needed to develop validated CAI.
- An opportunity to critique existing CAI.
- An opportunity to have hands-on experiences, both as a student via demonstration CAI programs and as a developer/author of a short CAI segment.

Given over a ten week period with four in-class hours per week, the CAI course is both an intro survey course and an applied lab course. Eight strands of instruction and practice continue through the ten weeks.

1. Removal of the mystery and terror from the process of computer access.

While there are many computers on campus ranging from micros to large time sharing main frames, the course participants use an IBM 1130/1500 system for most of their hands-on experiences. The course is designed such that all computer interaction and CAI demo occurs during class time, hence, each participant needs guaranteed individual access to the computer while remaining near a seminar/classroom area. The IBM 1130/1500 offered us this capability. Group demonstrations on microcomputers and other systems are given during class time by meeting temporarily at other locations on campus or by having vendors in.

2. History and applications of CAI.

3. Characteristics and potential of the medium.
- The content in these two areas is conveyed via

lectures, assigned readings, reviewing current literature, films, and audio tapes from conferences such as this one.

4. Evaluation.

The most common criticism of CAI is aimed at the quality of the computer's interaction as if the computer itself decided on the extent of instruction, the acceptability of the student's responses and the appropriate feedback. The author, not the computer, is usually responsible for bad CAI.

5. A programming language.

Coursewriter II is the language taught because of the use of the IBM 1130/1500 system for the students' CAI projects. There is a valid question here about putting so much effort into learning a language that is so dated and has such limited application. In answer, these are the facilities available to meet our needs. Secondly, in learning one's first computer language, one learns how a computer 'thinks' and the function as well as the format of the commands. This knowledge and experience makes a second language (perhaps a more practical language associated with a student's future coursework or employment) an easier one to learn.

Only the basic operation codes are taught. The first week students learn how a program is structured with labels and sequentially numbered statements; the second week, display, pause, and erase commands; the third week, commands necessary for a keyboard interaction; the fourth week, commands necessary for a lightpen interaction; the fifth week, explicit branching commands, counters and switches. By the sixth week, the students are well into coding their own CAI programs.

6. CAI project development.

The students follow a model of systematic instructional development. Flowcharting is taught to aid in the documentation of instructional strategies. The most difficult aspect of the project is keeping it short enough. Firm advisement in the first couple of weeks can avoid frustration and overcommitment near the end of the quarter. Students should also be advised that their projects will be developing at the same time that their awareness of CAI is increasing. They will not be able to foresee possibilities or have time to incorporate everything they may want into their projects. The projects value as an applied lesson and as a personal achievement outweighs the inevitable roughness of a first attempt.

The project is due to be debugged and ready to run in the tenth week. Classmates serve as willing and interested, although admittedly not always appropriate, test subjects. A formative evaluation report based on the test run and documentation on the program as it exists are due at the final exam session.

7. CAI demonstrations on a variety of systems.

At this time we are able to demonstrate CAI in other languages on a large time-sharing system (Xerox Sigma) and a small microcomputer (model varies). We also demonstrate PLATO, TICCIT (via slides only) and SIGI (a career counseling application).

8. Sources of information on CAI.

Professional and commercial journals, magazines, newspapers, newsletters, conferences and workshops.

Other recent descriptive and research literature.

Sources of CAI.

Vendors.

Objectives for the course are as follows:

CAI Application Objectives

1. Discuss in a one page essay the potentials of CAI comparing what the medium is capable of in the future with what has been achieved to date.
2. Describe (in one paragraph each) six types of computer applications in education.
3. Describe (in one paragraph each) four problems disadvantages or historic misuses of CAI.
4. Describe, in a one page essay, characteristics of a hypothetical teaching/learning situation that would make CAI an appropriate instructional medium.

Literature Objective

1. Submit one page abstracts for five articles, each article from a different journal or book. All articles must be at least as recent as 1975.

Hardware Objectives

1. Describe the configuration of the student terminals used in this course.
2. Given a diagram of the IBM 1500 Instructional System used in this course, name and describe the functions of each component.

Flowcharting Objectives

1. Given the basic flowcharting symbols, correctly describe the function of each.
2. Given a word description of an instructional strategy, flowchart it.
3. Given a short printout of a Coursewriter II program frame, flowchart the coded strategy.

Language Objectives

1. Sign on a designated unit in student mode, proceed through an instructional sequence using the light pen and the keyboard to make responses and sign off.
2. Sign on a designated unit in author mode and execute a given program sequence.
3. Given the 16 basic Coursewriter II op codes, write a description of the function and format of each code.
4. Given the strategy flowchart for an interactive frame, correctly layout and code the frame using the Coursewriter II language.
5. Given a short printout of a Coursewriter II program frame which has been edited (marked) for changes, sign onto the course in author mode and correctly input the marked changes.
6. Given a short printout of a Coursewriter II program frame which includes only the basic operation codes and a coded planning guide, correctly debug the on-line frame to meet the specifications of the planning guide.

Writing a Program Objective

1. Develop a short CAI program following an instructional development process including: statement of objectives
task analysis
flowchart
development of criterion items
scripting

coding
debugging
student run
formative evaluation
documentation

The CAI course has gained among IT students a reputation of hard work. A lot is expected in a short time. But its class roster is full every time the course is offered.

Graduates of the IT program are unlikely to gain employment in CAI centers (there are very few of them) or assume positions with responsibility for CAI but the students of this course will have an understanding of CAI on which to base their processing of further information on the subject. They will understand the jargon of the CAI field. With further support they could contribute to CAI development and/or application. They have put theory into practice and have had first hand experience with an increasingly more widely applied instructional medium.

THE COMPUTER IN CARTOONS:
THE SATURDAY REVIEW, 1946-1976

Kaye Reifers
The University of Mississippi

Walter M. Mathews
The University of Mississippi

ABSTRACT

Thirty years of computer cartoons from *The Saturday Review* are analyzed using common sense, journalism, and sociological techniques to determine the perceptions of cartoonists about the development and growth of computers in the American society. These perceptions are divided into four major themes: the computer is seen as an amusing toy, an anthropomorphic appliance, and impending malevolence, and a symbiotic pervasiveness. Each of these themes is reviewed and explained using slides of the cartoons as illustration. This analysis uses the perceptions of cartoonists about the computer as a means of assessing public opinion. In order to understand the future emotional impact of the computer, the past must be evaluated because this is a part of the evolving concepts of the public toward a machine which is becoming more prevalent in each American's life.

By definition, a cartoon is a caricature, often satirical, representing public opinion or attempting to affect public opinion about some matter. Cartoons have since their inception in the early 1800's presented a commentary on current events (Hess, 1969). Historically, the most potent cartoons have been the political ones. Within the area of politics, cartoonists have influenced as well as been influenced by public opinion. In other areas of our society, cartoons have also provided insights into the social climate and have influenced and even solidified the thinking of their readers about important social issues.

Americans have always had a progression of technology, with each innovation being a major focus of the American way of life. One of these is the computer. Cartoonists have used the computer as a major source of material and have accurately reported society's perceptions of the computer. One magazine which is highly regarded for its selection of cartoons is *The Saturday Review*. This magazine, through its editorship, has presented a valuable collection of cartoons depicting the cares and concerns of American society. Since the advent of the computer, the magazine has carried cartoon commentaries on societal perceptions of the computer, which include the economic and psychological impact of the computer, its pervasiveness and the enormity of its potential, as well as the evolution of the computer from an

amusing toy to one of the most awe-inspiring and misunderstood tools of humankind's technological development.

Sociologist Paul Baker suggested that social issues can be explored and understood by using three different sets of perceptions: 1) through common sense observations as used in everyday life 2) through the eyes of journalism, and 3) through the use of sociological techniques (Baker, 1979). In this presentation, thirty years of cartoons from *The Saturday Review* are examined by using, at times, each of these methods. The review of cartoons indicated that there are four major themes or perceptions that the cartoonists depicted: (Transparency 1) the computer was and is viewed as 1) an amusing toy, 2) an anthropomorphic appliance, 3) an impending malevolence, and 4) as a symbiotic pervasiveness.

Theme 1: The Computer as an Amusing Toy
(Transparency 2)

The invention of ENIAC in 1946 corresponded to the beginning of a period of rapid change in the American way of life. Following the first few years of its invention, the computer was not taken seriously. This lack of seriousness can best be seen by comparing computer cartoons of those early days with those of two other contemporary developments: television and the atomic bomb.

The cartoons of the atomic bomb are frightening: they each depict certain doom, the prevalent fear of the American public at that time. Americans quickly understood the technology of nuclear energy and its connection to their fears of eminent doom (slide 1). The atomic bomb was a constant threat to the security of life and continuation of the human species, as can be clearly seen in this cartoon of the children practicing in an atomic bomb drill (slide 2). The ease with which a war could begin is shown in this 1954 cartoon of the simple-minded politician who could destroy civilization with the touch of a finger (slide 3).

The other contemporary technology which captured the public's eye was television. This device had been developed much earlier but was just beginning to be manufactured for general use in the post war years. Once the television industry began the distribution and sale of its product, Americans quickly integrated this device into their lifestyle. Cartoons, depicting the advent of television, all emphasize how the common man's lifestyles were affected by the use of this device (slides 4 and 5). Even more importantly, the cartoonists envisioned the effect of television watching on the mores and behavior of future generations of Americans (slide 6). Both cartoonists and the general public were quick to understand the social implications of the widespread use of television, and the reflection of this is apparent in the cartoons themselves.

In contrast, those few cartoons in the magazine at this time depicting the computer presented an entirely different set of expectations about the future of this invention. The computer's influence was shown as affecting a far distant future in which little understanding was portrayed about the technology of how the computer worked or the potential of what the machine could do. They were viewed as mere toys that might someday help humankind take care of its dirty and inconsequential work. In 1946 the computer was viewed as an unnecessary tool that could do the messy jobs unskilled labor ordinarily performs (slide 7.) Then in 1951 the computer was ridiculed as a silly invention which had no importance in processing information, an unnecessary simplistic machine with which humans could play (slide 8). By 1954 the computer became viewed somewhat seriously as a tool that could handle our dirty jobs, but on a more sophisticated level (slide 9). The computer as an amusing toy is still a part of the evolving perceptions of the American public. Today, however, this view is a reflection of the intentional use of the computer as a recreational device for adults and

children.

Theme 2: The Computer as an Anthropomorphic Appliance (Transparency 3)

In theme two, an anthropomorphic appliance, the computer is portrayed by the cartoonists as exhibiting human traits, both physically and emotionally. In 1946 the cartoonists depicted the computer as a robot and in later caricatures, the human image was merely interspersed with more mechanistic looking devices. Not only through appearance, but also through behavior, the computer expressed human needs (slide 10) of parenting . . . (slide 11) of mating . . . (slide 12) of procreation . . . (slide 13) of needing a place in the events of human history. Between 1954 and 1957 expression of this type was very common. In 1958 another aspect of the human condition became prevalent in the cartoons: the psychological character of the computer became evidenced (slide 14). The cartoonists evinced the machine as exhibiting human behaviors which were considered negative or aversive. These traits included (slide 15) mindless subordination . . . (slide 16) depression . . . (slide 17) aggressiveness . . . (slide 18) disloyalty . . . (slide 19) protest. Throughout this time the computer was not considered a threat to the American public even though the machine was viewed as having negative personality characteristics similar to that of humankind. The average man's perception of the computer can best be typified by this cartoon showing the computer as (slide 20) "a little old man doing endless calculations."

But as the sixties dawned, fewer and fewer cartoons appeared in which computers were shown as personalities encased in humanoid bodies. A second aspect of the computer-human condition began to be shown. In these cartoons there is an evident feeling of hostility by the machine toward his human makers and masters (slide 21). The aggressiveness of the machine manifests itself in its demand and judgmental repudiation of the men in charge of the machine (slide 22) or those that are its unwitting victims (slide 23). The machine throughout this time shows quite clearly that not only can it be punitive, but people have no means of expressing their frustration, since it is beyond the control of human sanctions (slide 24).

During the seventies, another character aspect of the computer emerged. In this time period, the computer was shown manifesting the same types of behaviors that were becoming issues for social concern in the general population: (Slide 25) the computer experienced

hopelessness about humanity's future . . . (slide 26) the computer showed a need for searching to find the meaning of life . . . (slide 27) the computer sought pleasure above work . . . (slide 28) and the computer began shirking the responsibilities of adulthood.

An important analysis of this theme must include an evaluation of the people who were depicted in the cartoons with the computer. The most obvious observation of them plainly shows that only males are used. No cartoon shows a woman as a worker in the computer world. Not surprisingly, all or a large majority of the personality traits reflected on the computer are normally considered masculine in nature. The computer-human condition should perhaps be labeled only computer-man.

Theme 3: The Computer as an Impending Malevolence (Transparency 4)

In the early sixties the use of the computer became widespread. As its use became prevalent in the United States, the cartoons in *The Saturday Review* began to portray the fears that Americans felt about the computer. The earliest fear and the quickest to surface was that of losing one's job. The cartoonists depicted computers as a replacement for an individual (slide 29) and later as a replacement for larger numbers of employees (slide 30).

Just as the fear in response to computers in the work world emerged, so too, emerges another emotional response of man to machine. This was expressed as frustration (slides 31 and 32). The person dealing with this supermachine was automatically deemed inferior to the omniscient and omnipotent power of the computer (slide 33). The victims of the computer's errors or demands often reacted quite angrily, insisting upon dealing with the machine itself (slide 34). The people who worked with computers were depicted as nonentities, blindly following orders (slide 35).

Another major fear that surfaced was the progressive depersonalization of the individual until his life is listed as a series of numbers (slide 36). The cartoonists also showed the humans who work with the computer as dejected, lonely men in need of human touch (slides 37 and 38). These people were shown as having lost the warmth found only with other human beings.

The third major fear which surfaced was a two-fold one: 1) the fear of the awesome potential of the machine and 2) the fear that he might not understand how to control this awesome power. Examples of the power of the machine can

be seen in the cartoons which emphasize the self sufficiency of the machine (slide 39) . . . the sheer mathematical genius of the machine (slide 40) . . . and the incredible speed of the machine (slide 41). The second aspect of this fear is shown in man's lack of understanding of both the physical working of a computer and the actual method of processing information into meaningful knowledge for them to use (slides 42 and 43). The 1969 cartoon perhaps best illustrates how the average man felt about the computer, its products of overwhelming numbers and information, and his fears of the computer's impending malevolence to his emotional well-being (slide 44).

Theme 4: The Computer as a Symbiotic Pervasiveness (Transparency 5)

Again in the early sixties another theme appeared which depicted man and machine becoming increasingly complex in the development of the relationship between the two. Man appeared to become increasingly dependent on the machine, and the machines became growing monsters which expanded into all aspects of humanity's universe. As the computer became more pervasive, it was shown as involved in our ability to conduct war (slide 45) . . . in our development and our interpretation of the arts (slides 46, 47 and 48) . . . in the all-American pastimes of football (slides 49 and 50) . . . and dating (slides 51, 52 and 53). Perhaps one of the widest areas of influence of the computer is in its invasion of our language (slides 54 and 55) . . . large numbers of people began to use this new terminology in order to express their ideas in a clear manner (slides 56 and 57). The computer became a recognizable part of many aspects of our lives as can be seen in cartoon concerning revered institutions such as business (slide 58) . . . banking (slide 59) . . . in medicine (slide 60) . . . at home (slide 61) . . . in privacy (slide 62) . . . in education (slide 63 and 64) . . . in guilts (slide 65) . . . and in neuroses (slide 66).

Summary

This review of computer cartoons found in *The Saturday Review*, 1946-1976, indicates that there are four major perceptions or themes into which the cartoons fall. As perceived and portrayed by cartoonists, the computer became viewed as an amusing toy which had potential as a labor-saving device in the far future. The evolution of the computer has resulted in the computer actually becoming an amusing toy for the adults and children of a world thirty years later.

The concept of the computer, possessing human traits and character as well as somewhat human in form, was an idea in which the cartoonists evolved over the thirty years period. The principal features of the personality development of the computer were 1) human needs and psychological development in the forties and fifties 2) a feeling of hostility between man and machine in the sixties and 3) a machine that reflected the numerous ills of the seventies man.

Theme three and four both were born in 1962, hence their development simultaneously which ties people's fears about the computer to the dependency which Americans have developed. Theme three deals with the principal fears that the general public felt were threatening them. The cartoonists portrayed these fears: principally a fear of losing one's job, fear of losing one's individuality and becoming a number, lonely and dejected, and the fear of the computer's overall awesome power. Theme four deals with the invasion or pervasiveness of the computer as it wove itself into each strand of the American society and the symbiotic relationship between man and machine that developed.

BIBLIOGRAPHY

Baier, P. J. Invitation to Social Awareness: An Introductory Essay Comparing Common Sense, Journalism, and Sociology. Project on Developing and Evaluating Social Awareness Modules, Illinois State University, January 1979.

Hess, S., and M. Kaplan. The Ungentlemanly Art: A History of American Political Cartoons. New York: Macmillan, 1968.

COMPUTER LITERACY FOR 4th, 5th, AND 6th GRADERS

Sam Ewing
Adrian Public Schools

This paper describes the development, implementation and evaluation of a computer literacy program for 4th - 6th grade students. The program combines the use of time-sharing terminals and micro-computers to provide a three level computer literacy program. Over 1000 students at the 4th - 6th grade levels will be involved during the 1978 - 79 school year. The response from teachers and students has been extremely positive. The program appears to provide a reasonable model for implementation at the elementary, middle school/Junior High levels and possibly at the Senior High level. Brief course outlines, software costs, hardware costs, management techniques and record forms are included to assist classroom teachers and/or building administrators in developing and implementing a Computer Literacy Program for their school.

For the past 2 years, students and teachers in Adrian Public Schools have made significant use of instructional computing. Each of the buildings having 4th grade and up students has at least one Decwriter II on-line to the Lenawee Intermediate School District Time-Share System. The instructional terminals are in Media and/or Learning Centers in each building. Students and/or teachers are typically scheduled on a weekly basis in each school. The heaviest users in Adrian have been in Math Drill and Practice, CO-PILOT programs, simulations, and preparation of materials by the computer for off-line usage with students. The time-sharing terminals were supplemented this fall (1978) by a P.E.T. micro-computer purchased specifically for 4th - 6th computer literacy.

The addition of the P.E.T. has enabled the computer literacy project to get into 32 classrooms this year for a one week period. This represents 86% of the eligible classrooms in Adrian schools. In fact, all teachers except 6 from the 4th - 6th grade level have requested the P.E.T. for a week in the first year of implementation. This represents phase I of the computer literacy program. (See course outline)

Phase 2 combines a series of once a week lectures (5) with planned usage of a time-sharing terminal (HP2000) of 1 - 2 ½ hour sessions per student group (4 - 5) per week. This segment of the program uses My Friend - The Computer ©1976, by Jean Rice, as a textbook. This book provides an excellent introduction to computers for elementary students (4th - 6th). (See course outline)

Phase 3 provides students an opportunity to write BASIC programs. Both the P.E.T. and the H.P. Computer are used for this phase of the program. A Guided Tour of Computer Programming in BASIC ©1978, by Dwyer and Kaufman, is used as a basic guide for this portion of the program. (See course outline)

COURSE OUTLINES

I. Phase 1 (900 students)

Introduction to Computers
for 4th, 5th, and 6th Graders

- A. Micro demonstration
- B. Micro remains in classroom for one week with assorted programs
- C. Demonstration/Lecture - "What is a Computer?"

II. Phase 2 (90 students)

Computer Literacy I

(5th and 6th) Five ½ hour Lecture/Demonstrations (one per week) along with planned computer usage on HP (1 - 2 sessions per student group per week).

Uses My Friend - The Computer, by Jean Rice, ©1976

- A. What is a computer?
- B. How are computers used?
- C. History of the computer
- D. How does a computer work?
- E. Devices used to get information into and out of a computer.

III. Phase 3 (30 students)

Computer Literacy II

Introduction to Programming in BASIC

Uses A Guided Tour of Computer Programming in BASIC, by Dwyer and Kaufman, ©1978

- A. System Commands
- B. LET, PRINT, END
- C. GO TO, REM, INPUT
- D. READ/DATA
- E. IF - THEN
- F. FOR NEXT
- G. BASIC FUNCTIONS - INT(), SQR(), RND(), TAB(), ABS()

HP GET/RUN/LIST/SCR/SAVE
PET LOAD/RUN/LIST/NEW/SAVE

STUDENT REACTIONS

Natalie
I like your machine.
I played a dot game
I hope that you
will bring it back
because I like it

Reilly C. Dec. 19, 1978

The P.E.T. was fun to have
in our room. It is good to
help you with your math. I
like the Remorse Stand best
of all the games that you
can play on that. I would like
to get the one of that Pet
machine some day. Hope to
see it again. B+

12/17/78
why I like the P.E.T. because it
is fun and it has a lot of
good games and it plays
a lot of math games and
also other fun games
and that's why I like the
P.E.T.

I like the pet computer
because it plays math and
I think Dot 3, Math games;
drawing and lots more so
I hope you like it as much
as we did. I wish we could
get one for our room.

*Justin
Kellie
Reese*

COMPUTER USAGE

Request/Record

Dwyer/Kaufman. A Guided Tour of Computer Programming in BASIC, Houghton - Mifflin Company, Boston, MA, 1973.

Student Name	Grade	Date
<u>Program</u>	<u>Activity</u>	
1. \$DRILL1	Math Drill	
2. CENTI,A877	Lengths	
3. *HURKLE	Co-ordinate systems	
4. *LANDER	Land on Moon	
5. *PINUP	Makes Posters	
6. *SNOOP1	Snoopy Poster	
7. *SNOOP2	Snoopy Poster	
8. *STARS	Number guessing	
9. *LETTER	Alphabetical Order	
10. *OREGON	Oregon trip (1840's)	
11. *COP	Activities	
(COPilot)	LESSON?	
12. \$VOCAB	Spelling Practice	
	Hangman	
	Scrambled Words	
	What Spelling Unit?	

COMPUTER MATERIALS

Request/Record

TEACHER	ROOM	DATE
1. \$VOCAB	Word Search Puzzle	
	Spelling Unit?	
2. *PUZZLE	Word Search	
3. DRILLA,A877	Math Facts +	
4. DRILLS,A877	Math Facts -	
5. DRILLX,A877	Math Facts x	

Conclusions

Teachers and students feel computers are interesting and easy to use. The positive response from teachers and students means we will be asking for additional micro-computers for next year to extend and enhance the computer literacy program in Adrian Public Schools. The Computer Literacy program has lead to greatly increased use of computers for instructional purposes.

BIBLIOGRAPHY

Hatfield, Larry. Arithmetic Teacher, Volume 16, Number 6, pg. 53-55, February 1979.

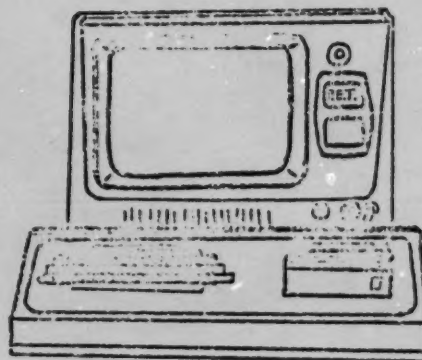
Rice, Jean. My Friend - The Computer, T.S. Denison and Company, Inc., Minneapolis, MN, 1976.

COMPUTER LITERACY

An Introduction to Computers for 4th, 5th, and 6th Grade Students

Session I: Introduction to Computers, "How to Operate the P.E.T." (30 - 45 minutes)

P.E.T. Micro-computer will remain in your classroom for you and your students to use for 1 week.



Session II: Demonstration-Lecture, "What is a Computer?" (30 - 45 minutes)

Please call extension 279 to arrange a week (12:00 - 1:00 p.m.).

**No more than 3 students per day. Each student may come a maximum of once a week.

COMPUTER USAGE PASS

Day/Date	Time:	Students:	Hour:	Time Returned
Mon/	_____	_____	_____	_____
Tues/	_____	_____	_____	_____
Wed/	_____	_____	_____	_____
Thurs/	_____	_____	_____	_____

Please send pass to Skill Center - will be used for Center Records

A COMPUTER LITERACY MODULE FOR JUNIOR HIGH SCHOOL STUDENTS

Ione Goodman, Steve Wetherbee, and
Daniel Neaton, John Marshak

Ann Arbor Public Schools

A computer literacy module designed for junior high school students was developed to inform students about the nature of computers and their impact on daily life.

Four math teachers, from four of the five junior schools in Ann Arbor, designed the module during the summer on a limited budget.

Students were pre- and post tested on attitudes toward the computer. The conclusions of the study are: students' knowledge of computers improved, and a computer literacy module should be included for all junior high school students in the system.

When we use the phrase "Computer literacy" what do we mean?

Literacy according to the dictionary is "The ability to read and write." People who are illiterate are limited in their every day functions thus ignorance of computers will render people as functionally illiterate as do ignorance of reading, writing, and arithmetic.

The development of a computer-literacy course at the junior-high level was recommended by the Conference Board of Mathematical Science sponsored by the National Science Foundation as long ago as 1972.

Much has been written about the problem of "computer literacy" and that schools must solve this problem by teaching it in the schools.

Teachers can go out and buy a reading course for teaching illiterate students but there is no course that can be purchased for teaching students computer literacy.

Four junior high school math teachers in Ann Arbor, Michigan spent their summer vacation designing a computer literacy module. They're objective was to inform students about the nature of computers and their impact on daily life.

The Committee had three summer meetings. Film and books were viewed on computers. Items were discussed that could be elements in the module such as tours, speakers, classroom games, computer attitude test, and independent projects assigned to members which would be included in the unit.

The main information resource for the module would be films and a text book. All items necessary for teaching the module would be put in a box and rotated around the schools. Items to be included would be: films, games, text book, study sheets for text book, tour suggestions, and a computer attitude test would be used as the "feedback agent."

This module would only be the "beginning" of a computer literacy course that could be modified as conditions changed in the computer world.

Proposed Outline of Computer-Literacy

Module for Ann Arbor Intermediate Schools

Day 1 ATTRACT INTEREST

Pre computer attitude test recorded on data cards

Activity: Read Pages 1-27 in textbook: Be A Computer Literate
Fill out completion sheet based on these pages in the book

Content: Introduction
What are Computers?
Discussion of two basic types of computers.
Study parts of computers.

Assignment: Unit Project e.g. Collection of Articles from newspapers and magazines.

Day 2 SOME TECHNICAL ASPECTS

Activity: Read pages 28-59 in Be A Computer Literate
Fill out completion sheet based on material read in the text book.

Content: Communicating with the Computer
Language of the computer.
How to write a simple program.
How computers work for us.

Assignment: Write alphabet, numbers their binary

Day 3 "HANDS ON" EXPERIENCE

Activity: $\frac{1}{2}$ class see film: "Discovering Electronic Music" (23 minutes)
 $\frac{1}{2}$ class "hands on" CAI and computer games
(Reverse at mid-period)

Assignment: Read selected history pages in text book for Day 4

Day 4 MINI COMPUTER DEMONSTRATION

Activity: Demonstration of a PET or APPLE II

Discussion of Reading of Day 3

192 Day 5 GOING BEYOND TODAY

Activity: Films: "Permutations" (7 min)
"Growing" (7 min)

Post Computer Attitude Test

Students turn in Unit Projects

(The textbook used was "Be A Computer Literate" by Marion J. Ball and Sylvia Chapp. It may be ordered from Creative Computing Press; P.O. Box 789-M; Morristown, New Jersey 07960.)

The actual modules taught in the fall by the four teachers varied greatly:

Steve Wetherbee's module outline for the Middle Years Alternative (MYA) Program at Forsythe Intermediate School follows:

COMPUTER LITERACY UNIT(MYA)

Day 1: Number systems other than base 10

a. Counting MOD 12 using clock as example

b. Base 5 using one hand and new names for the digits:

Base 10 1 2 3 4 5 6 7 8 9...

Base 5 1 2 3 4 10 11 12 13 14...

Name UN DU TRE FO FI FI-UN FI-DU FI-TRE FI-FO..

c. Talk briefly about Base 2

Day 2: Base two and logic

a. Writing numbers and doing arithmetic in Base 2

b. Work sheet on number systems (start in class, due Day 3)

Day 3: Class Functions as a "Computer"

a. Class sets up as binary to decimal converter using kids as flip-flops

b. Practice sorting four-bit cards

c. Solving problem in binary classification of trees

d. Input and sort students by four characteristics (hair color, eye color, glasses, sex)

Day 4: Films on computer graphics and hand out "Be A Computer Literate" books to be read during remainder of hour

Day 5: Do worksheets on "Computer Literate" book

Day 6: Introduction to "Basic" and a discussion of computers in general

a. System commands used: NAM
SAV
RUN

b. Statements used: PRINT
LET
INPUT
IF THEN

After this point, students had the chance to load simple programs and try out their skills (2 weeks were needed to give all 37 students who asked a chance to do this)

Also, the "Computer" using kids as flip-flops was set up again from time to time to try other sorting tasks and finally as a digital adding machine (adds numbers 0-15 to each other to obtain sums up to 32.

Ione Goodman's module outline for Tappan Intermediate School follows:

COMPUTER LITERACY MODULE (TAPPAN)

Day 1: Film: "What is a Computer?"
Pre Computer Attitude Test

Day 2: Two film strips from "Living with Computers":

1. Computers Can't Think (10:27)

2. Programming (7:03)

Students filled out completion sheets based on film strips

Unit Project was announced: Each student was to collect ten newspaper or magazine articles on computers in a notebook or folder. These were to be turned in along with all completion sheets and other materials from the module in two weeks.

Day 3: Read pages 1-27 in "Be A Computer Literate" textbook.

Complete work sheet based on reading.

Day 4: ½ class "hands on" the computer game of Blackjack (120 students played)

½ class see film: "Discovering Electronic Music" (23 min)

(Reverse at mid-period)

Day 5: Read pages 28-40 in "Be A Computer Literate"

Complete work sheets based on reading.

Two film strips from "Living with Computers"

1. Putting the Work In (9:17)

2. Getting the Work Out (5:33)

Complete worksheet based on film strips

Day 6: Two films: "Permutations" (7 min)
"Growing" (7 min)

Read pages 40-58 in "Be A Computer Literate"

Complete worksheet based on reading

Day 7: Two film strips from "Computers Can't Think"

1. Garbage In - Garbage Out (4:56)

2. For Good or Bad (7:42)

Complete work sheet based on film strips

Day 8: Guest speaker on History of Computers (Parent of student who teaches computer science at U. of M. School of Business)
Students saw vacuum tubes, relays, storage discs, etc.

Day 9: Post Computer Attitude Test
Student Evaluation of Unit
Review sheet on material for test

Day 10: Turn in Computer notebooks
50 question test on computer module material

Name _____
 School _____
 Class _____
 Date _____

GOODMAN'S COMPUTER OPINIONAIRE

DIRECTIONS: Please write your name, school, class and date in the upper right corner. Read each of the following statements and rate how you feel about it according to the following scale.

- Circle "1" no way.
 Circle "2" no.
 Circle "3" so-so or sometimes.
 Circle "4" yes.
 Circle "5" definitely yes.

- | | |
|--|-----------|
| 1. I like playing games on a computer. | 1 2 3 4 5 |
| 2. Computers confuse me. | 1 2 3 4 5 |
| 3. Computers make every day life easier. | 1 2 3 4 5 |
| 4. Computers cost too much. | 1 2 3 4 5 |
| 5. I would like to have a computer in my home. | 1 2 3 4 5 |
| 6. I am afraid I might break a computer. | 1 2 3 4 5 |
| 7. When I am on a computer, time seems to fly. | 1 2 3 4 5 |
| 8. Computers are too difficult to operate. | 1 2 3 4 5 |
| 9. I do not like to read about computers. | 1 2 3 4 5 |
| 10. Computers make boring matter easy. | 1 2 3 4 5 |
| 11. Computers are instructional. | 1 2 3 4 5 |
| 12. Computers control our life. | 1 2 3 4 5 |
| 13. Computers are too complicated for me. | 1 2 3 4 5 |
| 14. I do not want to miss my math class when I know I can get on a computer. | 1 2 3 4 5 |
| 15. Computers are taking many jobs away. | 1 2 3 4 5 |
| 16. I would like to know more about computers. | 1 2 3 4 5 |
| 17. Computers are dumb. | 1 2 3 4 5 |
| 18. When I grow up, my job will have a lot to do with computers. | 1 2 3 4 5 |
| 19. Computers are fascinating. | 1 2 3 4 5 |
| 20. Computers are making life too complicated. | 1 2 3 4 5 |

COMPUTER LITERACY, SCHOOL
MATHEMATICS, AND PROBLEM SOLVING
(3 FOR THE PRICE OF 1)

Fred Bell
University of Pittsburgh

Joint research and development efforts between the University of Pittsburgh and Upper St. Clair High School near Pittsburgh indicate that it is feasible to integrate computer literacy and high-level problem solving with the school-mathematics and college curriculums without sacrificing mathematics education programs. Incorporating computers and computing into high-school mathematics courses and college teacher-education courses had no negative effects upon students' achievement levels and attitudes toward mathematics and had a positive effect upon their attitudes toward computers. We found that integrating problem-solving and computer literacy with the standard school or college mathematics programs can aid in resolving some of the problem-solving, mathematics education, and computer-literacy "crises" in American education.

Ever after a 20-year period of phenomenal innovations in computer technology and computer applications in mathematics, and other school subjects, the long-awaited computers-in-education revolution has not yet arrived. The many and varied uses of computers in schools, especially in mathematics classes, has had an evolutionary effect upon teaching and learning in schools. However, the changes that have taken place in schools due to computer technology have been rather modest. What has been demonstrated in schools and research-and-development projects is that computers can be used in education to improve upon those things that were being done fairly well without computers. Computer-based tutorials, demonstrations, games, simulations, etc. have helped large numbers of students learn a variety of subjects.

In fact, computers have had enough effect upon education in particular and society in general so that Andrew Molnar sees computer literacy as "the next great crisis in American education" (THE Journal, July/August 1978). And this great crisis also subsumes three modest crises that are reflected in the following questions:

Can computers drastically improve mathematics education which, as reflected in the "back-to-basics" movement, has a crisis of its own?

Should we not be teaching high-level heuristic strategies for solving problems in school classrooms, especially in mathematics classes?

How can computer literacy be incorporated (even integrated) into the school curriculum without taking something else out?

Since these problems existed 15 years ago before computers came to school, and still exist today, how can we expect computer-enhanced education to accomplish in the future what it failed to accomplish in the past? The answer to this question may be found in the recent improvements in microtechnology, which resulted in the personal-computer concept. Personal computers are now a fact and distributors are not able to meet the demand for them. Computers were in industry, then came to school, and now are in the home. Once computers become as common in homes as TV sets, school people (teachers and administrators) will be forced to invite them into schools. The large majority of teachers who are presently ignoring the impact of computers in education, both as a learning aid and as a new literacy, can no longer do so in the near future.

The Crisis in School Mathematics

Scholastic Aptitude Test scores in mathematics are falling; students who can't even multiply and divide whole numbers are graduating from high school; only a minority of students and adults are able to do simple consumer-mathematics problems; and the back-to-basics movement is gaining followers. After 20 years in which billions of dollars were spent to develop the "New Math" and teach it to teachers and students, is it possible that there still is a crisis in mathematics?

ics education?

The nature of the current crises in comparison to the previous crises that led to "New Math" lies in the fact that the "New Math" failed to meet the expectations of its developers and promoters, more so than in the conjecture that things are worse now than they were before Sputnik scared us out of our scientific and technological wits. Many of the mathematics-education problems of the nineteen fifties are still with us as we prepare for the nineteen eighties. Mathematics literacy for everyone is still a problem but now everyone is more aware of it than people were in the nineteen fifties.

The Crisis in Problem Solving

Recently there has been a renewed interest in the nature of human problem solving and ways to teach problem solving, which is due in part to the interest in developing "intelligent computers." Many mathematics teachers and researchers in mathematics education now realize that what some people used to call problem solving, teach as problem-solving skills, and do research about was really exercise solving. That is, people were teaching very specific algorithms for solving very limited classes of textbook-type problems such as mixture problems, work problems, and rate-distance-time problems. Now people better understand, again due in part to research in machine intelligence, that general problem-solving skills are much more complex than simple algorithms. So the problem (or crisis) is finding ways to help students learn how to do "real" problem solving, which requires the use of complex cognitive processes in formulating unique cognitive structures.

Students must go beyond knowledge and understanding and do analysis, synthesis and evaluation. This means that students must not only get the answer but must also formulate problems, refine problem statements, gather information, analyze information, synthesize solutions to problems, and evaluate alternative solutions and solution strategies. We know through examples and experience that people can learn to carry out these and other activities in problem solving, but we don't know how to teach problem-solving skills and heuristic strategies to students in school.

The Crisis in Computer Literacy

Several satires have been written to illustrate how the development of reading and writing literacy would have been grossly retarded had it been carried out with the same constraints that are placed upon computer literacy. However, it is historical fact that the values of reading literacy and writing literacy were not

universally perceived, even in the United States until the twentieth century. It is also a fact that the majority of the populace of the United States are computer illiterates. However, since computers now have such a pervasive influence upon all of us, even though many people don't realize it, the need for universal computer literacy is becoming obvious to more and more people. Also, the TV, newspaper, and magazine horror stories about "computer" goof-ups are giving way to more positive (for computer literacy) stories about the valuable and interesting uses that are being made of computers. 195

While it is not at all obvious how computer literacy will be brought about, or how extensive it will be, the need for it is difficult to challenge. If we don't learn about computers and how to use them for good purposes, the "computer masters" may use them for their own purposes -- good or bad, with neither our involvement nor awareness.

A Joint Solution for All Three Crises

Math literacy, computer literacy, and problem-solving literacy have enough in common to suggest a joint solution for all three crises. One of the major problems in helping students become mathematically literate in school is lack of student motivation. Fortunately, the computer has been shown to be an excellent motivational device for large numbers of students in many mathematics classes, as well as being a good math tutor.

If computer literacy is to be taught in school, how can it be fit into the curriculum? By eliminating something else or by adding it on? Adding even one more required course is a very difficult task for most schools, which already have crowded curriculums. It is equally difficult to eliminate solidly-entrenched required courses or attractive electives to make room for something new. So one solution that suggests itself is to make computer literacy part of another course, several courses, or even many courses. Sliderule literacy, which is now of little use, was traditionally part of a math or science course whenever it was taught in school.

Even though the best way, or even a good way, to teach problem solving has yet to be found, it is generally thought that lots of practice is an important step in developing problem-solving abilities. Writing a computer program to solve a problem is itself a problem-solving situation because good programmers carry out all of the activities needed for problem solving that are mentioned above. Consequently, when students use a computer to assist in formulating a solution to a mathematics problem they get a double

196 benefit because they practice both mathematics and computer-programming problem solving.

Therefore, a modest joint solution for the math, computer, and problem-solving literacy crises is to integrate computer literacy in the school mathematics curriculum and in so doing develop better problems for students to solve than those typically found in school mathematics textbooks.

Resolving the Crises: An Example

Since 1970, I have been working on these three crises with John DeBlassio and James Saunders at Upper St. Clair High School near Pittsburgh. John is chairman of Upper St. Clair's Mathematics Program and Jim is director of its Computer Center. During this eight-year period we have developed an informal joint university/high-school program for integrating computers and problem solving in both the University of Pittsburgh Mathematics Education Program and the Upper St. Clair Mathematics Curriculum. Our work at the University involves very modest use of Pitt's 2 DEC system-1099 computer systems and total use of a recently purchased TRS 80 computer with 16K of core. Upper St. Clair has a PDP8/E system with 32K of core, 8 time-sharing terminals, and a card reader. This system is exclusively dedicated to student and teacher use in the 2,000-student high school and middle school. Nearly all of the people using the computer are mathematics teachers and students in mathematics. However, other teachers are encouraged to use the computer in their courses, but very few have elected to do so.

One undergraduate course or one graduate course (both were formerly called Computers in Mathematics Education) is required of all students in the Mathematics Education Program at Pitt. Since students majoring in other fields of education have recently shown interest in taking these courses, we have modified the content and changed the titles to Computers in Education. Consequently, these two courses are our approach to helping prospective teachers and experienced teachers begin to develop computer literacy. In both courses students learn BASIC, write a number of BASIC programs, and develop a computer-enhanced learning module that could be used by school students. Part of each course includes a half-day visit to Upper St. Clair to see what "real kids" in a "real School" are doing with computers. Problem solving and using computers to aid in learning problem solving are emphasized in both courses.

At Upper St. Clair, nearly all students become computer literate, to various degrees, through the hands-on use that

they make of computers in their middle-school and high-school mathematics courses. Rather than messing around, going to study hall, or visiting the library during free periods many students go to the computer center to write and run programs of their own choosing or to do computer-based math homework.

At the University, we used two methods to evaluate the success of the joint math, problem-solving, and computer literacy courses -- student evaluations of the two courses and the jobs that our graduates take when they get either a bachelor's, master's, or doctoral degree. Course evaluations are quite good but, more important, most of the computer/mathematics teaching positions in Pittsburgh and suburban schools are filled by people who have been in our programs. Several of the programs developed and implemented by these teachers have been cited as being exemplary applications of computers in education.

At Upper St. Clair, DeBlassio, Saunders, and I conducted a one-year, carefully-controlled research study of the attitudes, achievement levels, and other characteristics of school students in the computer-enhanced mathematics program. We found that the use of computer-enhanced resources throughout an entire algebra 1 course had no negative effects upon students' achievement in algebra, attitudes toward mathematics, and attitudes toward the instructional setting. During the study, students had a significant improvement in their attitudes toward computers and developed a respectable amount of computer literacy. We also found that students in the mathematics program at Upper St. Clair who liked using a computer in their math classes and learning about computers tended to have the following characteristics: favorable attitudes toward the instructional setting, high achievement in both mathematics and computer programming, high regard for solving problems requiring original thinking, and a high degree of vigor in school. Students who disliked using the computer tended to have these characteristics: poor attitudes toward the instructional setting, average achievement in mathematics and programming, average liking for problems requiring original thinking, an average degree of vigor, and a tendency to be responsible, cautious, and anxious.

In summary, our work over the past eight years does, in our opinion, provide strong evidence that integrating high-level problem solving and computer literacy with regular mathematics courses and programs can aid in solving the problem-solving, computer literacy, and mathematics education crises.

References

Bell, Frederick H. "Why is Computer-Related Learning So Successful? Educational Technology. December, 1974 (pp. 15-18).

Bell, Frederick H. "Can Computers Really Improve School Mathematics?" The Mathematics Teacher. May, 1978 (pp. 428-33).

Molnar, Andrew R. "The Next Great Crisis in American Education: Computer Literacy." THE Journal. July/August, 1978 (pp. 35-38).

Saunders, James. "What Are the Real Problems Involved in Getting Computers into the High School?" The Mathematics Teacher. May 1978 (pp. 443-47).

A COMPUTER SUPPORTED FIELD BASED TEACHER PREPARATION PROGRAM

Dr. John H. Schnabel, Professor
Southern Illinois University
Edwardsville, Illinois

A computer supported field based teacher preparation program is designed to provide an extended classroom teaching experience for the pre-service teacher on site at selected high schools. The placement of computer terminals, on-line to the central system, stand alone microprocessors or a combination of both at the field site will provide the pre-service teacher opportunity to get instructional help using computer instructional programs which deal with methodology, subject material, etc. and to use the computer technology in his/her classroom teaching with high school students.

Will teachers "miss the boat" of computer technology as they have with radio, television, and other supporting aids to instruction. Molnar states "the next great crisis in American education will be computer literacy." Teachers tend to teach as they have been taught. This statement has often been made. Experienced observation and analysis of classroom instruction will support it. Thus the importance of using the computer for instructional purposes becomes obvious. Teachers taught by computer and taught to use the computer for instructional purposes may assimilate the technology into their teaching style.

A number of questions arise when it is suggested that the computer be used for instructional purposes in the teacher preparation program. One of these stems from the fact that few public schools have computers or terminals connected to a computer system. The relatively high cost in the past of the equipment needed has been another factor--especially in times of financial stress. At the same time--during the periods of financial stress the layoffs of many teachers and the increased class sizes resulting points to the importance of an instructional support system such as the computer in order to maintain a quality instructional program.

Another question asks "Who will use it? No one on our staff knows anything about computers." A third question--"Do you mean I will be replaced by a computer" often comes from teachers. One answer to those questions lies in pre-service programs which help teachers learn the advantages and disadvantages of academic computing. The computer will not replace good teachers--it may change the perceived role of the teacher.

Declining enrollment in teacher education provides opportunity for assessment, redesign where desirable and the development of a quality approach.

In most teacher preparation programs University faculty spend a good share of time in the classroom teaching fundamental theories and expressing viewpoints about teaching in general. An additional amount of time is spent in the field with students in teaching learning situations. Students and faculty agree that there does not seem to be enough time to learn all the knowledge needed or to develop the skills at the desired level of competency.

Using electronic technology can relieve faculty of many of the repetitive tasks so that time may be utilized for other considered more important areas. Students may profit through spreading their learning time in required knowledge and skill areas over longer periods of time instead of trying to cram everything into a short two-year or less period.

Computer Assisted Instruction is one answer for developing a quality program in all areas and levels of Teacher Education. Based on the assumption that computer literacy will be an essential part of the teacher's life a program for preparing teachers that is supported by a Computer Based Learning System assumes major importance.

Further support of this assumption is the mounting sound of "Back to Basics" movement. The computer, with its endless patience is probably best fitted to meet the demands of supporting a teaching system for the basic skills of reading, writing, arithmetic, etc. Additional support of the assumption resides in the rapidly expanding technology of small computer systems purchased "off the shelf" and readily available in computer stores at prices within the reach of many students. The small computer system may soon be as much a part of many peoples' lives as the T.V. set.

One Way:

To avoid the next great crisis in American

education greater numbers of teachers will need to be involved in learning about the computer, how to use it, and how to develop instructional programs for it. One way to involve greater numbers of teachers in using the computer technology for instructional purposes is to include the computer as part of the teaching team preparing future teachers and to provide in-service training programs for current teachers.

It is not the purpose of this paper to discuss the field based part of the teacher preparation program. It is important to note however that the School of Education at Southern Illinois University, Edwardsville is beginning to combine the best efforts of computer supported instruction with a strong field based program in teacher preparation. In 1970 the SIUE field based program took prospective teachers into the public schools for a year of expanded clinical classroom experiences. University faculty traveled to the public schools to consult, advise and teach the prospective teacher in methodology, subject field strategies, classroom management skills, in attempts to combine theory with application in the classroom. Although the program has been eminently successful the time needed for travel between the University and school sites has gradually taken its toll in faculty energy and the elimination of desirable program elements.

In 1978 the School of Education Computer Laboratory was opened on a full time basis. Linked to the Mid Illinois Computer Cooperative the laboratory is equipped with Hazeltine CRT's, printing terminals and a graphics system. Instructional programs were shared by a number of friendly systems and the use of the computer for instructional purposes had its initial contact.

Concept of the Computer Supported Field Based Teacher Preparation Program.

The basic concept of a computer supported field based teacher preparation program is to provide an extended classroom teaching experience for the pre-service teacher on site at selected high schools. The placement of computer terminals, either on-line to the central system as stand alone microprocessors or a combination of both at the field site will provide the pre-service teacher opportunity (1) to get instructional help using computer instructional programs which deal with methodology, subject material, etc. and (2) to use the computer technology in his/her classroom teaching with high school students. In addition, it is anticipated that the in-service education for practicing teachers will be a plus benefit as they become involved with the pre-service teacher.

Importance of the Program

It is not clear at this time whether or not the use of the technology for instructional purposes will have any greater impact upon the students learning than previous methods of instruction. It is clear that the knowledge explosion continues at a tremendous rate. The computers ability to store and retrieve as well as manipulate information is a plus factor which can be

used to advantage. The rapidly expanding technology of the computer and its influence on nearly every facet of life as we presently know it puts computer literacy as an extremely important part of the educational process. The medical profession, business, industry and the armed forces are among the largest users of the computer for instructional purposes. Teachers must learn to use the computer in new and creative ways in support of the educational process. Failure to do so may result in non educators preparing instructional programs with learning activities taking place outside the educational system. Not that the latter would be all bad but may in fact be good--if teachers do not gain the expertise necessary to write desirable programs. The ready availability of low cost hardware is bringing the computer within reach of many persons for home and school use. It may be projected that as the hand calculator moved into the classroom so the microcomputer may make home education a real possibility.

Hypothesis

The hypothesis is that teachers who are taught to teach through a Computer Supported Field Based Program will have extended skills and knowledge for using the computer in the classroom.

Design

In designing a computer supported system for a field based teacher preparation program there are two distinct areas of development to be considered (1) faculty designed computer programs for the instruction of pre-service teachers and (2) programs developed by the pre-service teacher for use in the classroom during the field based experience.

Although teachers should have experience with and know how to use the computer for instructional purposes it is immediately evident that not all of them are interested in becoming computer specialists. The computer support program should be designed to involve--not immerse. For those who desire intensive computer training there should be the opportunity for enrolling in computer science courses.

Objectives

1. Identification of areas of knowledge and/or skills common for all teachers which can be taught by use of the computer at a level equal to or better than other methods.
2. The design and implementation of computer instructional programs in support of the Teacher Preparation system which will enable the pre-service teacher to demonstrate computer literacy through using the computer to learn and teach with at a predetermined level of success.
3. The design and implementation of those computer instructional programs which provide the pre-service teacher the opportunity to use the computer to achieve a success level in the knowledge and/or skill material to be used in teaching

equal to or better than by using other methods.

Role of the faculty member

One of the prime needs is defining the role of the faculty member. What will be the extent of involvement in designing programs? Faculty members are the chief ingredient in making the system work. In most cases the University faculty member had had little or no prior contact with using the computer. Orientation of the faculty to the advantages and disadvantages of a computer based instructional program is a first step in locating those individuals who will be interested in becoming involved. At the same time a certain amount of computer literacy will be taking place among all the faculty involved.

The current fact that writing of computer programs for instruction has not yet achieved the status of writing a book is a deterrent to many individuals who are seeking recognition for promotion or pay purposes. The freeing of faculty from teaching loads to engage in the design and writing of programs needs early attention.

Three Groups of Faculty to be Considered

There are three groups of faculty to be considered in the computer supported field based program for teacher preparation. The first group are those education faculty who are most involved in the methodology and theoretical aspects of teaching. The role of this group may vary from being an active participant in the developing and programming of computer materials to that of only a user of the computer program.

The second group is the subject field specialist. This individual will be working closely with the pre-service teacher in developing teaching materials for use in the classroom.

The third group--and one of the most often overlooked--is the supervising teacher in the public school where the field experience is to take place. This person can either provide the incentive for expert use of the computer in the classroom or can be a strong deterrent. Early involvement of the supervising teacher, the subject field specialist with the pre-service teacher is most important.

Developmental Stages

The field based portion of the SIUE teacher preparation programs has been in operation since 1970. With modest modification it will probably continue in the same pattern. Staging the development of the computer based portion of the program has required a number of special considerations.

Stage I:

Identifying faculty to participate in the development of computer programs is an ongoing situation. Education faculty are among the first to become involved. Subject field faculties are in the process of setting up orientation meetings.

Stage II:

1. The identification of basic knowledges and/or skills needed by all teachers. This area requires the expertise of all faculty, preservice and in-service teachers. Determining where the weaknesses have occurred in the past and how the use of the computer can be used to strengthen those areas needs input from all sources possible.
2. Decisions have to be made concerning the competency levels desired and/or required in each of the knowledge and skill areas.
3. Upon reaching agreement on items 1 and 2 determination needs to be made as to which of the areas can be taught as well or better by the computer than by other methods.

Programs that are in use or have been proposed for development in the process area include:

1. The writing of Instructional Objectives
2. Blooms Taxonomy (levels of learning)
3. Motivational skills
4. Planning the lesson structure
5. BASIC programming language
6. Using Tutor as a programming language
7. First year Teaching Simulation
8. Questioning Skills
9. Testing Procedures (how to develop good tests for evaluation purposes)
10. Instructional Analysis systems

and a number of others of similar nature. It is easily recognized that the early recommendations for use of the computer includes those time consuming repetitious teacher tasks which will free the faculty/teacher to become engaged in desirable humanistic efforts.

In the current computer library there are extensive programs written in most of the subject matter fields that may be drawn upon for examples. However, it is hoped that the pre-service teacher will soon develop new and creative programs designed for his/her own use.

Stage III:

As agreement is reached upon the basic knowledges and skills needed by all teachers and the desired level of competence to be achieved, the writing of new materials to be put into the computer will begin. This will be the basic text for the computer supported portion of the preparation program.

The following procedure will take place at this point:

1. Faculty will write material to be put into the computer program. This is basic text.
2. Basic text material will be reviewed by persons related to the areas, then routed

3. If a faculty member desires to program his/her own text and has the skill or with a minimum of help can do so this will be encouraged.

tion program into full operation will be three to five years. 201

Stage IV:

Testing and Evaluation

As each computer program is developed it will be tested for reliability, validity and documentation developed as to its use. A gradual implementation will permit students to be phased into the computer supported portion of the program with the least amount of confusion. Students and faculty will share in the test phases of each new computer program.

Testing the Hypothesis

How well the computer is used for classroom instruction will be revealed by an analysis of the programs used by the pre-service teacher in the field experience portion of the program. The analysis should reveal the purpose of the use of the computer--for individualized instruction, for testing or other purposes--and the amount of time used on the terminal during class periods or for other assignments. An analysis of the lesson plans developed by the pre-service teacher should show he/she sees the computer as to its many uses. Essentially, if a computer (either an on-line terminal or a microprocessor) is available for use and the teacher has the skill to use it then it should show in the lesson planning.

Time Plan for Southern Illinois University

A few Education faculty are involved in learning about the use of the computer for instructional purposes at the present time (Winter Quarter, 1979). Orientation programs for subject field specialists are planned for the Spring Quarter, 1979 to determine interest among those faculties. Students who will be involved in the program during the '79/80 school year will be identified during the Spring Quarter. During this same period the high school teacher who will be supervising during the field experience will be identified.

During the Summer '79 session it is anticipated that the high school supervising teacher and the pre-service teacher will start planning for the kinds of material which may properly be used during the field experience with high school students and which may be programmed for the computer. Some preliminary programming may take place during this period.

The standards for the anticipated amount of time available to the high school student for using the computer will need to be arranged with the school involved. This brings into focus the added factor of orientation of the school administrator and in some cases the school board members into the world of instructional computing.

It is estimated that the time to bring the Computer Supported Field Based Teacher Prepara-

- Academic Computing Directory. "Exemplary Institutions using Computers for Learning and Teaching" Human Resources Organization, 1978 Alexandria, VA
- Anderson, David. "Computer Aided Reading Instruction" Educational Technology, 1970
- Baker, Frank. "Computers and the Classroom" New York University Quarterly, Summer, 1978
- Bitzen, Donald, et al. "Computer Assisted Testing and Guidance" Conference Proceedings, University of Texas, 1970, Harper and Row, NY
- "Computers Are Fun, but Can They Teach" Educational Testing Service Developments, Winter, 1979.
- Conduit. State of Art Reports, Iowa City, IA, 1974
- Dennis, J. Richard. "Training Pre-Service Teachers to Teach with Computers" AEDS Journal, Winter, 1978
- Dyer, Charles A. "Preparing for Computer Assisted Instruction" Educational Technology, 1972.
- Goldenberg, Naftali. "Incorporating the Computer into the Technological Academic Yeshiva" T.H.E. Journal, September, 1978
- Johnson, Clemens M. Educational Uses of the Computer Rand McNally, Chicago, 1971, p. 239
- Levien, Roger Eli. Instructional Use of the Computer in Higher Education McGraw Hill, NY, 1972, p. 882.
- Margolin, Joseph B. and Hirsch, Marion. Computers in the Classroom Sparton Books, 1970, p. 882
- Molnar, Andrew. "The Next Great Crisis in American Education: 'Computer Literacy'" AEDS Journal, Fall, 1978
- Nasman, Leonard O. "Will Technology Revolutionize Education" Phi Delta Kappan, December, 1978
- Norris, William C. "Computer Technology and the Bottom Line" AEDS Monitor, October, 1978
- Peckham, Herbert. "Computers, Confusion and Complacency" Pergamon Press, Vol. 1, 1976
- Sam, Diane, Lorton, Paul and Muscat, Eugene. "Computer Based Instruction in an Out-Reach Program" Journal of Computer Based Instruction, Feb., 1978
- U.S. Department Health Education and Welfare. Teacher Training in the Use of Educational Technology: Proceedings of the Conference Washington D.C., February, 1978

EDUCATIONAL PROGRAMS IN COMPUTERS AND SOCIETY AND COMPUTER LITERACY^{1,2}

Richard H. Austing,
University of Maryland

Gerald L. Engel,
Old Dominion University

Abstract:

Over the past four years, the Association for Computing Machinery (ACM), with support from the National Science Foundation, has conducted a study of computer and society and computer literacy courses and materials.

In this study, the literature resources supporting such course has been carefully analyzed, and course recommendations have been prepared for levels ranging from the pre-college through continuing education.

This paper will briefly review the findings of the project and concentrate on the course and program recommendations at the pre-college and college-general education levels.

INTRODUCTION

The need to present educational materials on computer impact has been increasingly evident. Two distinct, but related aspects of this educational problem exist. In one case it is necessary for the computer professionals to provide information to the public at large regarding what computers are, what they do, and the impact, both potential and actual that the computer applications may have. On the other hand it is quite important that the developing computer professional be aware of those areas in which his work impacts society, what problems his actions may cause, and what is his responsibility to society at large.

Thus on the one hand, those interested in education regarding computing must be concerned with making the public sensitive to computers and computing, and on the other hand such individuals must be concerned with insuring that the computer professional completing a degree program, be sensitive to problems of modern society.

Within this paper both of these problems will be considered. Much of the work reported is based on the results of an invitational workshop held in Williamsburg, Virginia in July 1978 under the sponsorship of the Association for Computing Machinery (ACM) to consider course work on computers and society and computer literacy.

In this paper we shall first consider the course on computers and society or computer appreciation that would be available for the non-major. We shall then consider the course on computer impact that would be directed to the major in computer science and related degree programs. We will then address the general structure of such courses and how they might be implemented in instructional programs at other levels. Finally, we will consider resources that are available to support such courses.

COMPUTERS AND SOCIETY COURSE FOR THE NON-MAJOR

These courses have the longest history, and certainly have been the subject of several texts. A particularly good summary of the goals and objectives of such a course is provided by Lee[1].

"Irrespective of one's personal position on the role of computers in society, it is indeed desirable that all college graduates in the coming years have a realistic, even though minimal, understanding of how computers work and how they may be directed to implement and maintain almost any desired social system. Consequently, the primary purpose of this course on computers in society is to give an elementary but sound fundamental understanding of how computers work, what they can do, what applications of computer technology currently exist or are now on research consideration, and the relationship of these applications to the role of man in society. Thus, the course is conceived as a citizen's social problems course in which much of the time will be devoted to documentation of the claim that society is undergoing a computer revolution and to illumination of this position by the presentation of several problem areas resulting from computer applications."

In our first pass at this course, aimed at the non-computing major, we determined that the overall goal of such a course should be the provision of sufficient understanding and knowledge of computer systems to qualify students completing the course as "informed citizens" relative to questions concerning computer uses, limitations, implications and potential threats. It may, in fact, be argued that this knowledge is a portion of that which we traditionally ascribe to the "educated person."

We next considered the content of such a computer literacy course, and while it was the subject of considerable debate, we felt that such a course must meet the minimal objectives of providing the student with (1) a conceptualization of computer

organization (2) concepts of information and algorithms, (3) historic perspective, and (4) a critical assessment of capabilities, limitations and implications including what uses of computer have developed in society, what are the potential uses, and what threats these uses imply.

It was with this general description that we entered the Williamsburg Workshop. The question of the course was addressed by a small working group named "College-General Education". The group concentrated on the development of a course which was called "A Survey of Computers and Society" to illustrate in the title, the nature of the course. As we previously noted, there have been many such courses taught and as a result the specification of the course was given in much more detail than some other possible courses.

Generally, the course envisioned is designed for non-technical freshmen and sophomores with little or no computer background. The course could be taught by a computer science department or other suitable department and would normally involve 45 contact hours carrying three semester hours or four quarter hours credit. The working group strongly recommended that the understanding of this kind of material should be a part of the education of every college graduate.

The rationale for the course includes (but is not limited to) the technical/information explosion and the corresponding expansion in computer usage, need for awareness in consumer life, and need for more informed citizenry in decision making. Course objectives, somewhat more specific than those given in our first pass, are as follows:

- (1) to acquaint the non-technical student with computers, what they are and what they are not.
- (2) to introduce computer usage by first-hand experience in writing and running simple programs.
- (3) to understand the fundamental concepts of how a computer works.
- (4) to introduce the use of logical processes by means of problem solving.
- (5) to become familiar with computer usage in several applied areas.
- (6) to analyze the societal effects of computer usage and study trends in legislation concerning major issues.
- (7) to become more aware of the potential of computers to help man to solve problems in a technological society.

The following catalog description for the course was suggested:

This survey course is designed to acquaint the non-technical student with computers and how they work, their uses, and the impact they are having on society. Students will gain first-hand experience in writing simple programs and running them on the computer, and become more knowledgeable about uses of the computer in such areas as; business, health care, transportation, education, and government. By analyzing related social issues such as the threats to privacy, increased productivity through automation, and the possible

loss of individuality, students will become more aware of the potential uses and abuses of computers, and can better serve as informed citizens in today's technological society.

A sample course outline would then be as follows:

I. Introduction

- A. Computers and Computer Science
- B. Overview of the Computer Age
- C. Nature of Computers and Computing

II. Communicating with a Computer

- A. The Access Language
- B. Problem Solving
 1. Algorithms
 2. Flow Charts
- C. Programming and Languages
 1. Using one Language
 2. Overview of Other Languages
- D. Software
- E. Documentation

III. Hardware

- A. Computer Organization
- B. Storage Devices
- C. Input/Output Devices
- D. Codes
- E. Computer Systems

IV. Applications by Area/Function

- A. Business
 1. Data Processing
 2. Management Information Systems
 3. Data Banks
 4. Electronic Funds Transfer Systems
- B. Government
 1. Data Processing
 2. Control
 - a. Traffic Control
 - b. Law Enforcement
 3. Planning
 - a. Forecasting
 - b. Simulation
 - c. Other Applications
 1. Health Care and Medicine
 2. Law
 3. Transportation
 4. Process Control
 5. Education
 6. Humanities
 7. Politics

V. Issues

- A. Privacy
 1. Intensification of Issue
 2. Data Dissemination Control
 3. Surveillance
 4. Legislation (world over)
- B. The Workplace
 1. Automation
 2. Skills Deterioration
 3. Job Roles
 4. Information Overkill
- C. The Individual
 1. Personalization/Depersonalization

2. Changing Role of the Home
3. Personal Computing

VI. Future

- A. Trends in Technology
 1. Microprocessors
 2. Distributed Computing
 3. Personal Computing
- B. Organizational Impact
 1. Organizational Structure
 2. Political Process and Decision Making
- C. Artificial Intelligence
 1. Robotics
 2. Game Playing
 3. Pattern Recognition
 4. Natural Language Comprehension
- D. Man-Machine Symbiosis

References are plentiful and the reader is referred to the Library List prepared by a joint committee of the ACM and the IEEE Computer Society [2] for a partial listing. Percentages have not been given for the time to be devoted to each topic to allow the institutions to adjust to their particular needs and objectives. It may also be noted that numerous materials exist besides textbooks for such courses such as periodicals, audio-visual materials, bibliographies and bibliographic reference services, related readings, guest speaker services and projects.

In spite of the fact that much work has been done on this type of course, there remains a number of open issues that need to be addressed. These include the following.

- A. Teaching schemes (pros and cons of one teacher or several teachers as a team; use of guest speakers, what areas might they address and how are they selected; class size; grading)
- B. Teaching methodologies (lectures; lab management of students; use of AV materials; use of computer equipment, what kind and when; use of CAI; use of homework)
- C. Vocabulary/concepts/skills (depth of technical material; depth of programming requirements)
- D. Activities (field trips, tours of campus or other facilities; projects).

COMPUTERS AND SOCIETY COURSE FOR MAJORS

This area represents the attempt, through course work in a major program, to make the student aware of the areas in society which his work may impact. Whether or not this should be a formally required course in every curriculum or included in a variety of courses has been the subject of considerable debate. The ACM Curriculum Committee in Computer Education for Management (C³EM) recommended such a course in its graduate recommendations [3] and in the undergraduate recommendations [4]. In the new recommendations of the ACM Curriculum Committee on Computer Science (C³S) for the undergraduate program [5], such a course is a strongly recommended elective, and it is suggested that it should be required if the course material is not covered elsewhere in the curriculum.

The underlying theme in all this work is a desire to instill in the major a concept of how his work affects society at large, and how in fact

it is necessary that the professional be knowledgeable about and make responsible decisions concerning the nature of the work performed.

205

In our first pass at this course, which we titled computer impact on society at that point, we noted that the goal is more involved than in the case of the course for the non-major. At the most basic level it is clear that we would like the students to be aware of the various impacts of computers on society. By itself this goal carries with it a strong implication that the knowledge gained by the student in the course will be merged with his own value structure to provide a base for professional judgements and actions which occur during his career. Alternatively, awareness may be seen to be merely the necessary and not sufficient condition. The sufficient goal would include providing the student with (1) concepts of social value and valuation, (2) models which describe the impact of computers on society, (3) a framework for professional activity which provides for explicit consideration of and decisions concerning social impact, and (4) tools and techniques which are applicable to problems posed by the social impact of computers.

In terms of the types of problems and needs that we face we may sight the analysis and design of business information systems as an example. Daily systems are implemented which produce an undesirable impact on the organizations they serve as well as an arguable impact on society in general in many cases. Within organizations, business information systems may affect employment, the existence and content of jobs, skill levels, managerial techniques, organization structure, distribution of power, security and communication patterns. Certainly this is not an exhaustive list, and we are not describing an exception.

It may be argued that the exceptional business information system is one which does not impact one or more of the elements listed above. Perhaps more important is that many of these impacts are not planned but rather occur after-the-fact as a response or adjustment to the implementation of new systems. Certainly some impacts are planned or may even be specific objectives of the new system design. Job content and employment effects are probably the most frequently planned. On the other hand, secondary effects such as power shifts and changes in communication patterns are rarely, if ever, planned.

Lack of planning of social impact, to whatever degree, is particularly inexcusable in the design of business information systems since we typically go through a process such as feasibility analysis where we carefully examine technological and economic feasibility. In fact, we argue that analysis and evaluation of social feasibility should be an integral part of the process of systems analysis and design.

Since feasibility analysis constitutes a special case and is not a part of all computer system implementations we have extended our argument to call for the inclusion of what we have described as impact evaluation to be included as

a distinct step in the implementation of any computer application. The course for majors, then, has to do with the preparation of the student to make such evaluation.

With this kind of material as background, a working group titled "College-Major" developed material for an upper division course on the social issues and impacts of computing at the Williamsburg Workshop.

The group developed a course proposal for an upper division, first year graduate course which assumes that most of the students have had some prior experience as computer programmers or users. As a result the course does not address the technical understanding which students need in learning the concepts developed in the course; these are assumed as prerequisite.

The major goals of the course, as envisioned by the working group, are to help students learn to carefully analyze the social settings in which computing is used; to understand what forces (social and historical) give rise to different system uses and designs; the impacts that such systems have upon different parties (both users and non-users); and the value conflicts raised by computing. It is the intent of the course design that the students who take such a course will develop an appreciation for analyses developed by others which are based upon different values. The course then provides an excellent opportunity for the major to develop his capacity to think critically.

Some of the specific principles that such a course should help the student understand are the following:

(1) Many of the problems raised by computing often occur when computer systems or the social setting in which they are used are relatively large in scale. (Large scale programs, particularly those that are ad hoc design are often hard for their designers and users to comprehend and alter to better serve their users. When computing is used in settings which include many individuals, groups, and organizations which share common systems, computing becomes an instrument among others, of intergroup conflict. Also many participants whose connection with a system, is limited to some specialized portions of it, may have trouble understanding the blend of organizational procedures and technical operations by which it works. These phenomena of large scale differ fundamentally from those one finds with small systems used only by one person).

(2) Computing is a socially problematic technology. (While computing is often introduced in organizations to help "solve" certain organization or broader social problems, computing often raises new problems. Computing may be viewed as a "problem transformer" rather than a "problem solver" when its use is embedded in complex social settings).

(3) Social and technical fixes don't work well. (While some of the social problems raised by computing may be reduced or altered with

"technical fixes", such efforts do not fundamentally resolve many problems. For example, many of the privacy problems raised by computing may be less troublesome if computer systems were technically very secure, but security provisions alone do not resolve these problems).

It is principles such as these, and certain methods for analyzing the roles of computing in complex social settings, that are what we wish students to learn by the end of the course. Principles such as these can be studied and given concrete meaning through the study of computer applications in certain areas (e.g. health care, education) or certain social issues (e.g. privacy, shifts of social power). This type of course proposal offers the instructor substantial freedom in choosing technologies (e.g. simulation), arenas, and issues to teach more fundamental principles or methods of analysis.

The course recommended should be organized into two major segments. At the beginning the instructor should introduce the analytical perspective which will be developed in the course and place the development, use and impacts of computing in a broader context. The instructor may wish to follow an analytical approach based on historical analyses and begin the course with a brief social history of technology. Alternatively, the instructor may emphasize the different values and interests held by different people and groups involved with computing and begin with an account of value position often encountered in analysis of complex technologies. Yet other faculty may prefer to emphasize sociological analyses and introduce technology in social settings for such a perspective at the beginning of the course. Faculty teaching such a course may adopt one of several different theoretical perspectives. It is critical that they make their perspective explicit to their students at the beginning of the course and indicate how that perspective may be used to help understand social issues of technologies, and then develop the course with cases drawn from computing.

It is quite difficult to specify a set of candidate topics and appropriate readings. More topics may be specified than can be covered in a meaningful way in one semester or quarter. Some of these topics are, of course, mentioned in the outlined for the non-major. It is the case, in fact, that not all possible topics should be covered since the primary value of this course is to educate students so that they may continue to learn and to understand the social issues and impacts of computing on their own. Thus the instructor should choose judiciously from the set of possible topics. No less than one week should be devoted to each topic. The collection should be sufficiently varied to include some variety in social scale (e.g. work place through larger scale technologies), issues (e.g. privacy, comprehensibility of systems), and arenas (e.g. education, health care). Any one topic could be developed further in more advanced seminars or topics courses.

Appropriate references for each topic must be determined and the quality and quantity of

of literature is somewhat limited. Though not as fully outlined as the previous course, this should supply some ideas regarding construction of such a program. Obviously the details are dependent on institutional and program objectives. Areas such as projects and pedagogical approaches (e.g. lectures followed by discussion, debates, case studies, exercise, etc.) need further consideration.

OTHER EDUCATIONAL PROGRAMS ON COMPUTER IMPACT

From the earliest point in the project it was our feeling that the two courses that have been discussed, with suitable modification, could be utilized for varying audiences from the pre-college level through continuing education.

The workshop results reenforced this belief. Early in the course of the workshop sessions, the participants were divided into three groups to address general educational issues. The result of this was the construction of a three-dimensional matrix consisting of topics, level and audience. The audience would include pre-college through continuing education. The level could be classified roughly as knowledge, attitude and skill, and the topics would include the following:

(1) Organizational Impact

- Survey of Applications
- Work and Leisure
- Electronic Funds Transfer and Financial Systems
- Privacy and Surveillance
- Computer Utilization/Underutilization
- Computers and the Future
- Artificial Intelligence/Man's Intelligence
- Computers and the International Scene
- Medicine/Health Care
- Computers and Education
- The Information Society
- Ethics, Social Responsibility, Professionalism
- Political Processes
- Personal Computing
- Law, Crime, Legislation
- Computers and the Humanities/Arts
- Computing Economics
- Simulation and Modeling

(2) Organizations

- Education
- Health Care
- Trade Unions
- Military
- Business

Education

- CAI, Administrative, General Classroom Use, CMI

Health Care

- Planning
- Evaluation
- Regulations

Work

- Employment
- Working Conditions
- Moral Issues, Quality of Work

Cultural

- Impact On Our Language and Speech
- Music, Art, Research (Qualitative and Quantitative)
- Library Use
- Values/Self Understanding
- History
- International Issues/Developing Countries
- Political
 - Impact on Voting and Politics
 - Political Process
 - Privacy, Legal Issues, Citizen Participation
- Computer-Communication Networks
- Electronic Funds Transfer Systems
- Community Information Utilities
- Business-Economy
 - Computer Industry
 - User Industry
 - Inflation
 - Economic Models and Forecasting
 - Gross National Statistics

(3) Social Areas

- Workplace
 - Automation
 - Skill Deterioration
 - Insufficient Tolerances
 - Information Overkill
- Home
 - Official Intrusions
 - Commercial Intrusions
 - Ombudsmen
- Education
 - Professional
 - General Use

(4) Changes

- Microtechnology
- Distributed Capabilities

(5) Logical Organization

- What is a Computer?
- Powers/Limits/Intelligence
- History
- Communicating With a Computer
- Programming
- Operating Systems
- Equipment
- Applying a Computer
- Areas
 - Art
 - Business
 - Health
 - Government
 - Sciences
 - Engineering
 - Planning
- Functions
 - Data Processing
 - Input/Output
 - Control
 - Computing
 - Simulation
- Impacts/Problem
 - Privacy, Regimentation, etc.
- Personnel
 - Functions
 - Careers

The construction of a particular class of courses then becomes a matter of filling in the cells of the matrix appropriate to the objectives of the particular program. As an example consider the results of the work of the "Pre-College" working group. The goal of the group was that there should be universal computer literacy. At the high school level, graduates would be considered computer literate if they have knowledge of the following areas:

- (1) Historical perspective of computing.
- (2) Computer anatomy (including parts of a computer, how computers work, algorithms, problem solving, systems capabilities).
- (3) Uses of the computer (both types such as information storage, simulation, data processing, and communications, and areas such as business science and technology, education, health care, etc.).
- (4) Social implications (such as careers, organizational changes, privacy, etc.).
- (5) Futuristics (for example, trends in artificial intelligence and robotics, innovation and new technology, communications).

It was highly recommended that students also achieve an introductory level skill in flowcharts and programming. Where computer equipment is available it was recommended that every consideration should be given to making this skill a requirement. On the other hand, the development of attitudes and value judgements is not a primary goal at this level.

This indicates some of the ideas of such course construction. Obviously a number of questions remain open and work needs to be done. Included here are the following:

- (1) Definition of content, including amount of detail and structure.
- (2) Approaches (for example hardware-free, familiar to unfamiliar),
- (3) Modularizing and sequencing for grades K-12,
- (4) An approach to developing a skill level in programming,
- (5) Teacher readiness
- (6) Identification of materials, especially those of high quality,
- (7) Coordination with international efforts at this level.

LITERATURE RESOURCES

In addition to our focus on courses, a major thrust of the project was to identify and organize literature on computer impact. The sheer volume of available material forced the development of an information storage and retrieval system to maintain it. This has resulted in an annotated bibliography which includes identification of each entry by activity, area, function, level, and type. Multiple classifications of each item and within each category are possible and retrievals may be made by various combinations of identifying information. The system currently contains about three thousand entries and will be available in hard copy form later

this year. Continued maintenance of the bibliography and maintenance for future special purpose searches has been insured by ACM.

Of special interest are trends shown in the literature. The literature identified may be roughly divided into the categories of (1) awe-inspiring or wonder-induced, (2) alarmist, (3) application descriptive, and (4) empirical prescriptive.

The earliest articles tended to show a preoccupation with computer capabilities relative to human thought and capabilities. This, of course, also lead to the alarmist literature, with alarm centering on questions of employment impact, dehumanization and the thrust of nationwide data banks. A corresponding body of material argued that computer systems do not pose a threat, or at least not a threat of the magnitude suggested by the alarmists. Alarmist literature has continued into the 1970's but drastically decreased in volume. Furthermore, material being published today tends to be directed more to specific problems and threats such as credit reporting systems, electronic funds transfer, or computer security.

Literature describing applications of computer systems which impact society is vast. This literature displays very little chronological grouping and is found throughout the period studied. A subcategory of application descriptive literature could be described as application analytical. In this material, the implications of specific applications and general application areas are examined at various levels of detail.

The smallest category is also the most recent to develop; the category which includes both the description of empirical studies of the impact phenomena and prescriptive material.

In an attempt to estimate the relative quantity of the various literature categories, a rough analysis was made of the bibliography. If the categories of awe-inspiring and alarmist are excluded on the grounds of lessened current interest and value, the remaining categories seem to break out as follows: application-descriptive 50%, application-analytical 47%, empirical and prescriptive 3%.

It should be noted that there is a lack of precision in these figures and a possible bias in the construction of the bibliography itself. The important point is, however, that even if we double our estimate of the number of empirical and prescriptive entries it is still a minor fraction of the total literature. Even more important is the point that the absolute number of these entries is also quite small.

The literature, then, relevant to the impact of computers and society consists of an almost trivial base of empirical information together with a massive array of analytical discussion of applications and offering only a minor amount of prescriptive material.

Turning to our original course materials, we find that the course for the non-major is well served by the literature. This course is primarily descriptive and roughly 97% of the material falls into this category. In contrast, the course for the major is not well served. The necessary goal of awareness is supported, however the sufficient goals cited are only minimally supported. The bulk of such materials must come from carefully prepared empirical and prescriptive materials and these by and large do not exist.

SUMMARY

In this paper we have presented some models upon which computers and society courses to meet several needs could be constructed. Such courses are essential and their development, at all levels should not be deferred.

To properly prepare and present such courses, requires a great deal of research and conceptual development. We require a workable, generally accepted taxonomy upon which we may organize our knowledge into useful forms. We also need much more empirical evidence concerning actual effects of the implications of computer systems. We need both more refined models which describe and allow the prediction of the impact of computer systems and we need new tools and techniques for dealing with the problems of computer impact. Perhaps most importantly, we need a framework by which results of research and development can be applied in a meaningful way to the implementation of computer systems, and may be adequately integrated into the curriculum.

NOTES

- (1) The work reported in this paper was partially supported by NSF grants MCS 74-00434 A01 and SED 76-84222.
- (2) Portions of this paper were adapted from "Literature Resources and Computer Impact on Society and Computer Literacy Courses" by Richard H. Austing, William W. Cotterman and Gerald L. Engel, appearing in the Proceedings of the Computer Science and Engineering Curricula Workshop, IEEE publication No. EHO 126-3, IEEE Computer Society, 1977.

REFERENCES

- [1] Hans E. Lee, "Computers in Society -- A Course Description, Purpose and Rationale," SIGCSE Bulletin 4, 1 (March 1972), 97-102.
- [2] A Library List on Undergraduate Computer Science, Computer Engineering and Information Systems, Association for Computing Machinery, New York, January 1978.
- [3] R. Ashenhurst (ed.), "Curricula Recommendations for Graduate Professional Programs in Information Systems," Communications of the ACM 15, 5 (May 1972), 363-398.
- [4] J. Daniel Cougar, "Curriculum Recommendations for Undergraduate Programs in Information Systems," Communications of the ACM 16, 12 (December 1973), 727-749.
- [5] R. Austing, B. Barnes, D. Bonnette, G. Engel, G. Stokes (eds.). "Curricula Recommendations for the Undergraduate Program in Computer Science," Communications of the ACM (March 1979).

EXPERIENCES OF "MY PAL THE COMPUTER"

IN THE PRIMARY GRADES

Jean Rice, TIES, St. Paul, MN 55113

Sandy O'Connor, Fridley Schools
Fridley, MN 55412

This computer literacy unit is designed to provide children in the primary grades an opportunity for exposure to the computer. An early introduction to the computer provides students a computer foundation from which to build in subsequent years. The unit evolved following several years of field testing various approaches to computer literacy in the primary grades. Changes were implemented in the program over the years as technology and applications changed. In particular, microcomputers have most recently been added to the program. Reactions to the program on the part of parents, teachers, administrators and students have been extremely positive.

INTRODUCTION

How do primary grade children develop a factual basis of information about computers? Are these young children capable of using a computer in a meaningful way? Should we introduce computers in the primary grades? The following report discusses a computer literacy unit developed for primary age children.

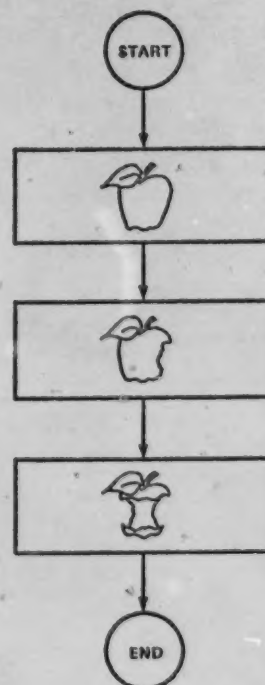
HISTORY OF PROGRAM DEVELOPMENT

Can computers clean up your dinner dishes or babysit for children? Can computers clean your room? These are some of the questions asked by children in primary classes when first introduced to the computer.

The unit begins by having the children build their conception of a computer using boxes, tempera paint, masking tape, paper fasteners, buttons, scraps of fabric, and the like. These "computers" usually turn out to look like robots. A list is then made of what the children thought computers are capable of doing. The children thought computers could talk, walk, babysit, destroy things, read and tell the truth if people are lying.

After this introduction, the children are introduced to flowcharting. Their writing vocabularies are much smaller than their speaking vocabularies. As a result, the children draw pictures in their flow charts instead of printing words. Since sequence is so important in learning to read, this activity helps the children develop logic patterns.

A flowchart of eating an apple might look like this.



The next part of the unit is spent on computer terminal keyboard recognition. The children practice finding special keys. A giant keyboard in the front of the classroom is used, and the children each have a smaller dittoed keyboard. They practice finding their names, numerals, letters and words like PRINT, END, RUN and SCR. They also use a regular typewriter to practice on during free time.

They are next introduced to programming and how to type in and run a program at a terminal. They write programs such as the following:

```

SCR
10 PRINT "AMY"
20 PRINT "I LOVE YOU"
30 PRINT "KING KONG"
40 PRINT "THE FONZ"
50 PRINT 2 + 2
60 PRINT 10 - 2
99 END
RUN

```

For second graders a slightly more advanced program might look like this:

```

SCR
10 PRINT "TROY"
20 PRINT "STAR WARS"
30 PRINT "I LIKE LUKE SKYWALKER"
40 PRINT "8 + 8=" 8 + 8
50 PRINT "15 - 7=" 15 - 7
60 END
RUN

```

The children's enthusiasm at the terminal is seen daily as they run their programs.

During another part of the unit, discussions take place concerning the various types of computers; how computers work much faster than human beings; how computers can only carry out the instructions for which they have been programmed and how computer programmers make mistakes not the computer. At the end of the unit, the list of what the children thought computers could do is checked against what was learned to see where revisions might be necessary. A filmstrip tour of a computer center is also shown. The children bring home the folders they keep of all the activities.

PRESENT DAY PROGRAM

The preceding described the unit as taught 7 years ago. Since these early years of teaching computer concepts, there have been many changes. Computers have become very visible. This computer unit has also undergone some changes. At the beginning of the unit, time is spent discussing where computers are used in traffic control, in outer space and air travel, at home in video games, personal computers, microwave ovens, at stores and for billing systems.

Computer morality is a current topic. Discuss-

ions take place on how some kinds of information are private. The identification number (ID) and password that is assigned when using the large computers are discussed along with the need to keep this timesharing code secret. It is mentioned that older brothers and sisters have been known to use these ID's and use up the storage and time assigned to other schools.

Volunteer parents are used to assist the students at the terminal while the teacher is involved with the teaching. This involvement of the parents provides them with a better understanding of their child's computer learning and results in a positive reaction to the program.

In addition to inputting their own programs at the terminal, students also spend time running programs that are simulations or that reinforce reading, spelling, language, and arithmetic skills. For example, the children are particularly enthusiastic about a spelling program that types a spelling word and then quickly blocks the word out. The child must remember the word and type it back. The terminal rings bells for correct responses.

Microcomputer demonstrations have recently been added to the unit. The addition of graphics, color, and sound throws a whole new dimension into our computer curriculum. A game of hangman shows a robot that is saved from hanging only when a child is able to spell correctly one of his weekly spelling words. In another program a car must cross a ravine. Each correct response to an arithmetic problem builds part of a bridge that will help the car across. The children are most enthusiastic with the graphics available on the microcomputers.

As a result of field testing the previous approaches to computer literacy and activities in the primary grades during these past 7 years, the computer literacy unit, MY PAL THE COMPUTER, has evolved. The unit includes a chapter on computer applications, a short history of computers beginning with simple counting and the abacus, through calculators, and ending with present day computers, large and small. Various kinds of computers and their parts are discussed. Simple flowcharting and BASIC programming are also introduced. There are activity sheets throughout the unit such as constructing a flow chart, doing a crossword puzzle of computer terms, and playing calculator games to name a few.

Because the unit is in story form and relates common day experiences that a child has with computers, the children can relate their daily experiences. Most have played a video game or have seen a microwave oven. They are surprised to learn a computer is inside of each one. They are anxious to convey this information to their parents. They have seen large computers on TV, and they have many questions about the flashing lights, the robot voices, the turning tape reels and the print-outs.

CONCLUSION

As computers are now a part of our lives, the need exists for all to become computer literate. Parents and teachers involved in this program had previously been apprehensive and expressed numerous fears about computers. Their involvement in the program calmed their fears and apprehensions. It became a very positive experience.

By starting computer education early, children do not develop these fears. They are confident when handling the equipment, and they learn computer capabilities and limitations. Computers are able to do many of the things children dreamed they could do; but the children realize people, human beings, make these actions happen. It is the feeling of these authors that children involved in these various primary computer literacy classes have been learning that we can make a computer work in almost any way we plan. It can be our friend--MY PAL THE COMPUTER.

AN IN-SERVICE MODEL TO PREPARE ALL
ELEMENTARY TEACHERS IN ONE SCHOOL
DISTRICT TO USE COMPUTERS

Dr. Gerard J. Putz
Macomb Intermediate School District

The Department of Instruction from the Macomb Intermediate School District (a regional service center) in cooperation with the CBI Coordinator from the Lakeview Public Schools system are preparing all elementary school teachers in the Lakeview School District to use computers for educational purposes.

The in-service training, released time for teachers, and the time sharing computers are provided by the Macomb Intermediate School District. The parent organizations purchased terminals. The costs to the school district are the cost of telephone lines and telephone jacks for each classroom.

Over 90 percent of the students surveyed after completion of the project agreed or strongly agreed that:

1. They enjoyed using and learning about computers.
2. The computer helped them with their math.
3. They would like to learn more about computers.
4. They would like to continue working with computers.
5. Learning about computers is important.

All of the elementary teachers rated their learning experiences with computers and the classroom applications as good or excellent.

The following goals were identified by the teachers from one school as priorities for the series of ten seminars over a three month period:

To learn how to access the Macomb Intermediate School District computers by correctly logging on through Beechwood Elementary's Decwriter II terminal, and to learn the mechanics of the computer terminal and coupler and learn basic computer commands (i.e., hello, run, list, bye, etc.)

To be able to define the appropriate language associated with teleprocessing (i.e., coupler, on line, terminal, ports, etc.); and learn how to use the introductory computer language (BASIC) to increase their ability to communicate with the computer, write elementary programs for student use and to teach students basic programming skills.

To experience and evaluate application in the area of Computer Assisted Instruction (CAI) which will include tutorial learning, games, simulations, testing and drill practice.

To complete one grade level of the Hewlett Packard Drill and Practice mathematics program and learn how to assign students to this program for record keeping purposes.

To learn how to use the Compute Program to assist them in diagnosing their students' needs and to generate worksheets for students in mathematics instruction.

To experience and evaluate existing computer games to determine the instructional appropriateness of these programs for their students.

To diagnose student learning levels in mathematics, provide students with alternate learning activities and maintain student progress reports through Computer Managed Instruction (CMI) practices to include an introduction to

Cincinnati Instruction Management System (CIMS)
and Macomb Teacher Support System (MTSS).

minimum expenditure of funds.

To explore and evaluate existing control library programs to determine the instructional appropriateness of these programs for their students.

To experience and evaluate existing computer simulations to determine the instructional appropriateness of these programs for their students.

To become familiar with the impact and importance of computers in society.

To have the opportunity to explore other computer related applications, such as Seeker IV, Guidance Information System, Michigan Occupational Information System, Readability Formula applications and Computer Based Resource Units.

The in-service education workshops dealt with these topics during ten - three hour seminars which included instruction in new applications, a practicum in a computer lab and a critique and evaluation of classroom problems and successes.

The success that this project has achieved is due to the insistence that computer use become an integral part of the classroom as opposed to a segregated computer lab, the use of a curriculum infusion process versus a mandatory participation process and the continued coordination provided by a knowledgeable person (CBI coordinator).

This project will result in the training of all elementary teachers in one district to use broad classroom computer applications with a

INTEGRATING COMPUTER CONCEPTS INTO COMMUNITY COLLEGE BUSINESS CURRICULA

Roy James Parrish, Jr.
Associate Professor, Data Processing
Danville Community College
Danville, VA 24541

The use of the computer has greatly influenced the day-to-day activities of most occupations. This statement is particularly true of business-related occupations. Community colleges and other two year post-secondary institutions offering programs of study leading to employment in business-related jobs must prepare their students to function in this new occupational environment. Most business programs recognize this fact and have attempted to provide the student with a knowledge of data processing concepts and procedures by the addition of an introductory data processing course--with unsatisfactory results. Danville Community College has undertaken an extensive effort to integrate computerized methodology with various business subject areas (i.e., accounting, statistics, management, finance, secretarial science, etc.). This integrated approach has proved to be superior to the traditional introductory data processing course.

Introduction

Most educators and employers would agree that the widespread usage of computers has significantly influenced the duties involved in the performance of most jobs. Persons employed in business-related occupations frequently discover they are dependent on a computing system for the day-to-day information needed to perform their job functions effectively. The increased usage of computing equipment in business occupations is of extreme importance to instructors in the business curricula of community colleges and other post-secondary institutions. Many graduates of business programs from these institutions gain employment in entry level positions in which they actively prepare data for entry into a computer system, maintain control totals or perform other data validation activities. If community colleges and other two-year post-secondary business programs are to provide instruction which will enable the student to enter the business world upon graduation and to function effectively on the job, the student must receive instructions on the use of the computer while he is enrolled.

Most business educators agree on the importance of providing the business student with an opportunity to gain at least an introductory knowledge of data

processing procedures and computerized methodology. Many institutions have attempted to meet the business student's need for a knowledge of data processing procedures by the addition of an introductory data processing course (i.e., Introduction to Data Processing or Principles of Data Processing).

The Traditional Method

As was previously stated, the traditional method of providing the business student with a knowledge of data processing concepts and procedures has been through a single introductory data processing course usually lasting one quarter or one semester. The content of this course has typically included such topics as: History of Data Processing, Punched Card Data Processing, The Data Processing Cycle, Components of Computing Systems, Numbering Systems, Programming Languages, etc. All topics listed are important topics if the student is to understand the importance of the computer in the business world. However, many, if not most, business students complete an introductory data processing course with little or no knowledge of the role of the computer in the day-to-day operation of a business organization.

There are obvious reasons for the failure of the introductory data processing course to provide an understanding of operational data processing concepts and procedures. The student enrolled in the introductory course normally has no background in data processing; therefore, course content must begin with the simplest concepts and progress to more complex topics. This presentation of basic data processing concepts will consume almost all the time allotted during a one quarter or one semester class leaving little time for explanation of specialized business applications. The use of projects in the introductory course may not necessarily improve student understanding of the role of the computer in business, because most projects are fragmented and lack relevancy to the student. The student is often more interested in completing the project by the due date than he is in understanding the significance of the activity in the business world.

An Integrated Approach

The community college or other two-year post-secondary business student should not be expected to gain an in-depth understanding of data processing concepts and procedures through successful completion of an introductory data processing course. At best, the student can be expected to gain a knowledge of basic data processing concepts (i.e., Computing System Components, The Data Processing Cycle, etc.). Previous statements are not intended to belittle the content of the introductory data processing course as the "basics" are prerequisite to more complex topics. The intent is to point out that our expectations for the introductory data processing course are unrealistic.

A greater understanding of the uses and role of the computer in the operation of an organization can be acquired if the student has an opportunity to experience the use of the computer in business subject areas. A discussion of the use of the computer in accounting in an introductory data processing class may have little relevance for the student because he is merely an observer. However, when the student uses the computer to solve accounting problems in his Principles of Accounting class, topics such as data accuracy become meaningful. The entire concept of computer usage in business becomes realistic because the student is actively participating in the process. Business curricula contain many courses whose content allows (if not requires) the integration of computerized concepts and methodology.

The Integrated Approach at Danville Community College

Danville Community College is a comprehensive community college offering programs in a variety of occupational-technical, university-parallel, and business areas. Enrollment consists of approximately 2300 day and evening students of which approximately 600 are enrolled in business programs. The Business Division of Danville Community College offers the associate in applied science (AAS) degree in the areas of data processing technology, business management, merchandise management, and accounting. An associate in science (AS) is offered in business administration as well as a certificate in secretarial science (1 year).

Business division faculty at DCC recognized the need for a knowledge of data processing concepts by business students years ago. Consequently, an introductory data processing course has been a requirement for all business programs, with the exception of business administration, for the past 15 years. It was also recognized that business students were not gaining sufficient knowledge of business applications from this course. During the past four years, extensive efforts have been made to incorporate computer concepts and computerized methodology into as many business subject areas as possible. Feedback from graduates, employers, and curriculum advisory committees indicates that students currently emerging from business programs possess a greater understanding of the role of the computer in business. Many evening students are already employed in positions which require computer usage. To these students, manual completion of problems, activities, etc. is completely unrealistic.

It should be pointed out that efforts to integrate computer concepts into business subject areas at DCC did not include the assumption that all students should be able to write programs. The underlying assumption was that most students would be users of computing services when they gained employment in business. A brief description of the manner in which computer usage has been incorporated into various subject areas follows:

Accounting. Extensive use is made of the problem-solving capability of the computer in accounting instruction. All students enrolled in Principles of Accounting solve their accounting problems using an instructional accounting packaged developed at DCC. The student codes accounting transactions, records these transactions on an input medium, and submits the job to the computer for

processing. The transactions are edited with the student correcting any keying or coding errors--just as would be required in business. When all errors have been corrected, a set of documents is printed. The student then analyzes the results of his output. During the Principles of Accounting course, the student becomes familiar with transaction processing, the importance of accuracy, and the data processing cycle.

The exposure to computerized processing of accounting transactions benefits the accounting major in auditing. The importance of an audit trail in computerized accounting systems becomes apparent when the student attempts to trace a transaction through the accounting cycle.

In addition to the use of the instructional accounting package used in Principles of Accounting and Auditing, computer usage has been incorporated into Payroll Accounting. The time required to calculate a small payroll can be exasperating. Programs have been developed at DCC which allow the student to code payroll transactions with the computer producing a payroll register. The student is able to concentrate on concepts without being overwhelmed by tedious calculation.

Statistics. Statistics students solve various problems using an instructional package of programs secured from the publisher of the statistics textbook currently being used. Students have the opportunity to use the computer to solve problems rather than performing laborious calculations for hours. The use of the computer as a computation tool leaves the student much more time to analyze the results of his calculations.

Finance. Computer usage has been integrated into the finance course through the use of a ratio analysis package of programs developed locally. This package again relieves the student of many tedious calculations. In addition, a relationship between the keeping of accounting records and financial analysis is established. This linkage of concepts occurs through the correlation of financial statement data produced by the package of programs used in Principles of Accounting with the ratio analysis package. Income statement and balance sheet figures produced by the accounting package become the input to the ratio analysis package. The relationship of accounting to other business functions is thus established.

Almost all business students at Danville Community College enroll in Principles of Accounting, Statistics, and Finance. Through these courses,

they receive considerable exposure to the role of the computer in a business organization. It should be noted that not all assignments in the courses listed are carried out on the computer. The student is required to be able to perform the activity manually in order to ensure understanding before it is carried out with the aid of the computer. 217

Problems

The obvious problems associated with this approach to integrating data processing or computer concepts into subject areas are attitudes, hardware, and software.

Attitudes. If the faculty member does not see the need for computer usage in his subject specialty, obviously, integration is impossible. A possible cause may be the instructor's lack of training in the use of computers. Integration of computer concepts into a subject specialty can only occur with the cooperation of the instructor.

Hardware. Integration of computer concepts into business subject areas will require the availability of computing hardware for student and faculty use. Funding for new or additional hardware is always difficult to obtain in educational institutions. One method of emphasizing the need for additional facilities is to initiate a project integrating computer concepts into a course using existing hardware. If the hardware is inadequate, students will take the message to the administration. Another possible consideration is the acquisition and use of a microcomputing system. Recent advances in technology have placed microcomputing systems in a price range affordable by most institutions.

Software. There are three possible sources for software to use in this integration procedure. These are: develop it yourself; buy it; or get it free from another institution. Regardless of the source, the following points should be considered when selecting software:

1. Ease of use. Any package selected should be as simple as possible to set up in a computing system. The ideal package requires limited knowledge of programming by either faculty or students
2. Simplicity of student input. The successful student usage of a package of programs requires that input requirements be simple. Any package selected should contain edits of input to facilitate corrections of keying errors.

3. Test student knowledge of subject matter. The package used in a course should require the student to demonstrate his knowledge of the subject matter. The student should not be assured of correct output merely because the input passes all syntax checks.

Advantages Derived from Integration

Providing the business student an opportunity to experience the use of the computer in business-related applications while enrolled will better prepare the student to function effectively when he secures employment. The "black box" image of the computer is dispelled, and the student better understands the role of the computer in the business world. Institutions preparing students for employment in business must prepare students to function in an environment highly dependent on the computer.

AN INTERDISCIPLINARY APPROACH TO COMPUTER SCIENCE

William W. Agresti
University of Michigan-Dearborn

Abstract

Computer science is viewed as an interdisciplinary study. First, various possible relationships among disciplines are traced, and the features of interdisciplinarity are explained. Next, a specific interdisciplinary program in computer science is introduced, including experiences with the administration and operation of the program. An appraisal of the interdisciplinary approach to computer science education follows. The advantages are significant: a lower cost program, a more adaptive program, and one that is better able to treat the interfaces of computer science with other areas. Such a favorable report suggests that the interdisciplinary approach merits consideration, especially by colleges which want to offer a computer science degree but have limited funds.

Introduction

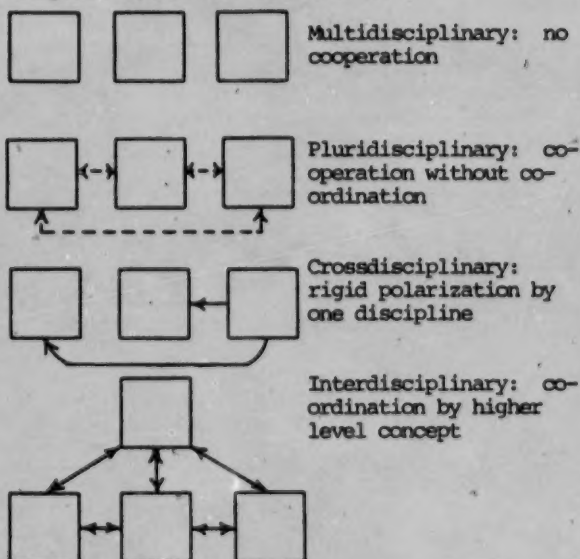
Undergraduate programs in computer science can be found in many different places in colleges and universities [1]. There may be a separate department; or the computer science program may be housed in a related department, often engineering, mathematics or business. Where it is located, of course, is dependent on the characteristics of the local environment - especially how such a program was initiated. What is discussed here is a third possible presentation of computer science - as an interdisciplinary program. Such an organization is not unique [5], but "user experiences" with an interdisciplinary approach should be of interest.

What is Interdisciplinary?

It appears that whenever two people from different departments talk, it is hailed as an "interdisciplinary" interaction! Actually, the presence of more than one discipline does not guarantee interdisciplinarity. The figure in the next column displays four relationships among disciplines: multi-, pluri-, cross-, and interdisciplinary [3]. The four are presented in order of increasing cooperation and coordination.

If we consider "computer science education in colleges" as an example, then instances of each type of organization can be given. In the multi-disciplinary case, the disciplines of business, engineering, and mathematics, for example, may be teaching computer courses, but there is no cooperation. The step up to pluridisciplinarity comes when the first signs of cooperation appear. To continue the example, this may take the form of simple bilateral agreements on the use of computer facilities or

permission to substitute one's introductory computer course for the other's.



Crossdisciplinarity arises when the principles and practices of one discipline are imposed on other disciplines so that the entire organization becomes polarized by that one discipline. Whether this can happen with the computer science education example depends on the personalities of the individuals and how evenly distributed the computer science people are throughout the disciplines involved.

Finally, we reach the interdisciplinary stage, which is distinctive because there is not only cooperation but also coordination. The coordination is guided by a sense of purpose, a unifying goal. The effective and efficient delivery of computer science education is such a unifying goal which can underlie an interdisciplinary approach.

Of the four relationships, an interdisciplinary one would exhibit cooperation and significant coordination with a sense of purpose. So, in terms of form, a computer science program could be constructed with those features. On matters of substance, the question is whether or not the subject matter of computer science can legitimately be viewed as interdisciplinary.

Because computer science is relatively new, it is easier at this time to identify parts of it as coming from more established fields, like electrical engineering and mathematics. Involving, as it does, significant amounts from (at least) mathematics, engineering, and business, computer science appears to be a subject that can be thought of as interdisciplinary. This view is consistent with most descriptions of computer science [2, 4], which stress its relationships to other areas of study.

The Program at The University of Michigan-Dearborn

At the University of Michigan-Dearborn (UM-D) by 1974, computer courses were being taught in the School of Engineering, the School of Management and the Department of Mathematics. Each group had faculty members with professional interests in computer science. The range of courses is evident from the list below.

School of Engineering

Introduction (FORTRAN-based)
Assembly Language Programming
Computer Organization
Simulation
Data Processing and Management Information Systems
Logic Design & Switching Theory
Programming Languages
Systems Programming

School of Management

Introduction (FORTRAN-based)
Programming Languages
Computer Organization
Information Systems

Department of Mathematics

Introduction (FORTRAN-based)
Computer Organization
Assembly Language Programming
Data Structures
Programming Languages

The course duplication was considerable - a situation which is especially ill-advised in times of tight university budgets. Also, the students didn't have as broad a range of computer courses to choose from, considering the number of courses being taught. There were enough "computer people" on the faculty to teach more varied courses but they were busy offering many versions of the same courses.

This was the situation when it was decided that a complete program should be introduced, leading to the Bachelor of Science degree in Computer and Information Science. An objective in the program design was that maximum use should be made of existing courses. The core of the new program was very nearly the "Curriculum-68" - with the course material naturally updated and somewhat different. Of the Curriculum-68 core courses, only Discrete Structures was missing from the list above and would be created specially for the new program.

Administratively, Computer and Information Science is a program, not a department, in a Division of Interdisciplinary Studies. The coordination is provided by a program director and a program committee of workable size (5) with representatives from each of the three units involved.

The obvious move at the start was to schedule the offering of the core courses to eliminate redundancy. The faculty members freed by these moves would be able to teach extra intermediate and advanced courses. Now that students are beginning to move through this program, these courses - like Compiler Construction and Theory of Computing - are being developed.

In its coordinating role, the program committee first decides what courses should be offered during the coming term. Next it determines what unit will teach each course. The committee's recommendations are sent to the heads of the units involved (the Dean of the School of Management, Chairman of Industrial and Systems Engineering, and Chairman of Mathematics). The unit heads are cooperative, and the recommendations have been well-received.

Advantages and Disadvantages of an Interdisciplinary Program

ADVANTAGES

1. Greater opportunity for "genuine" interdisciplinarity

The supporting faculty have backgrounds and interest in non-computer parts of the disciplines from which they come. As a consequence, there is better treatment of the interfaces between, for example, computers and business or computers and engineering. By "genuine" interdisciplinarity, I mean that the computer science program is distinctive because of this strength at the interface areas. Such an interdisciplinary program perhaps could do a better job of teaching the relationships between computers and other

fields. Opportunities for scholarly activity on such relationships may be enhanced.

2. Lower Costs

Organizing computer science education as an interdisciplinary program may be the most economical and practical way to begin to offer a computer science degree. There is less overhead by not having another department. The University of Michigan-Dearborn did not have to add any new faculty to begin its Computer and Information Science Program. The coordination of existing faculty was all that was needed. The unit heads benefit also because, if a course is recommended by the program committee, there is some assurance that the enrollment will be satisfactory. In the past, due to a lack of coordination, a low enrollment might arise because a similar course was being offered in another department on campus.

3. More Adaptive

The members of the program committee can be changed easier than in a departmental faculty. As a result, an interdisciplinary program would be more responsive to change - and with a rapidly developing subject like computers, this can be important. The administrative framework may make it easier to include new interdisciplinary areas - like computers and society of computer applications in the social sciences. In this way, an interdisciplinary program could avoid the tendency of some departments to become fairly static in terms of activities and interests which were built around the permanent faculty of the department.

Also, the distribution of computer science courses among supporting departments is fairly natural. While there are many ways to partition the subject matter of computer science, most are quite similar. It may be thought of as hardware, software, theory, and applications or, perhaps, systems, programming, theory and applications. If a program does admit to divisions like this, then the assignments below may fit.

Hardware (Systems)	-	Electrical Engineering
Software (Programming)	-	Industrial and Systems Engineering or Mathematics
Theory	-	Mathematics
Applications	-	the department of application, especially Management for business applications.

The inclusion of industrial engineering may not seem obvious. Actually such departments often have a close relationship with computer science (see, for example, [6]). The above list exhibits somewhat the current assignments at UM-D.

DISADVANTAGES

221

1. The interdisciplinary program is not at a higher level in the organization

To be effective in its coordinating role, the program should be at a level which is higher than the disciplines themselves [3]. Because the program is at the same level as other departments, the program committee makes recommendations and lobbies for their acceptance. It relies on the cooperation of the department heads to accommodate the needs of computer science education within their budgets. That the program is less institutionalized than a department is both an advantage and a disadvantage. The program is more tenuous than a department - more easily ignored or eliminated.

2. Divided Loyalties

As with most university examples of interdisciplinary programs, this one has the potential of divided loyalties of its supporting faculty. This has not been a problem at UM-D. The faculty have a strong interest in computer science and there has been a willingness to work for the success of the program.

Conclusions

The introduction of an interdisciplinary program in Computer and Information Science has been successful at the University of Michigan-Dearborn. Because of its significant advantages, especially economic ones, an interdisciplinary program should be considered as an alternative to other ways of organizing for computer science education.

Acknowledgements

Many past and present faculty members contributed to the development of the CIS program at the U of M-Dearborn, including Dr. Robert Lyjak, Dr. Thomas Richards, Dr. John Riordan, Dr. Victor Streeter, Dr. Alan Goldfine, Dr. Adnan Aswad, and Dr. Eugene Arden.

References

1. ACM Graduate Assistantship Directory in the Computer Sciences 1974-1975.
2. Anarel, S., Computer Science: A Conceptual Framework for Curriculum Planning. Comm. ACM 14 (June, 1971), 391-401.
3. Jantsch, E., Inter- and Transdisciplinary University: A Systems Approach to Education and Innovation. Policy Sciences 1 (Dec., 1970), 403-428.
4. Newell, A., Perlis, A. J., and Simon, H. A., Computer Science. (letter) Science 157 (Sept., 1967), 1373-1374.
5. Ward, D. L., A Cooperative Effort of a Computer Science Program. SIGCSE Bulletin 7,5 (July, 1975), 78-83.
6. Webster, D. B., and Zaloom, V. A. A Computer Science Option in Industrial Engineering. Proc. Third Symp. Computer Science Education, SIGCSE Bulletin 5,1 (Feb., 1973), 125-131.

THE INTRODUCTION OF CAI TO AN UNDERGRADUATE DECISION MAKING COURSE

Susan S. Lenker
Edward J. Fisher
Central Michigan University

ABSTRACT

The paper describes efforts at Central Michigan University to incorporate computer aided instruction in the teaching of a core course in management science in a undergraduate business program. Advantages of using CAI in teaching such topics as linear programming, network analysis, forecasting and regression analysis seem to include improvement in student attitude, consistency of homework format, the use of more complex and instructive problems, greater integration of decision making techniques, more motivational and conceptual lectures, greater uniformity among sections in covering necessary topics, the inclusion of additional important topics, and the training of students in the use of software packages to solve management problems.

INTRODUCTION

The School of Business Administration at Central Michigan University, like other Business Schools across the nation, is the fastest growing school in the University. The demand for our three credit management science course, I.S.A. 385 has grown from three offerings per semester in 1975 to the six presently offered. Recent changes in the core curriculum have been made including a requirement for all undergraduates to take I.S.A. 385. These two factors have led us to project a minimum of 12 offerings per semester as early as next year.

As a result of this increase, the composition as well as the interests and needs of the students have changed. Until now the students were comprised mainly of accounting majors and management majors who possessed a fairly good mathematics background. Standing on the threshold of change, we anticipate a shift in the interests and preparation of the general business student who will take the course. This student may no longer possess the more rigorous options of calculus and statistics but rather have a generalized year of algebraic techniques and a brief introduction to statistical concepts. (Increasing the math prerequisite for the general business student may take three to five years before becoming reality). Moreover, this student may not have as great a desire to enhance the quantitative skills any further as did former students taking the course as an elective. Fortunately, the Business School has been able to maintain a good working relationship with the mathematics department. As

a result, specialized topics such as linear algebra have been included in the prerequisite courses. The general student, however, may still display many preconceived and counterproductive attitudes about the course.

THE COURSE

The introductory quantitative decision making course consists of a survey of those mathematical techniques found to be most employed by business and industry [6]. Until this time, emphasis had been placed on the computational skills involved in these techniques. Since former students could achieve these goals with little difficulty, there was little incentive to change. The new students, however, are not as likely to be computationally oriented. A further impediment is the fact that most students know neither their future roles in business nor the types of skills they will be presumed to possess.

In order better to meet the needs of the general student, we are in the process of shifting the emphasis in I.S.A. 385 from the analytical "tools" themselves to the management context in which they will be used [1,2,3]. The survey of quantitative techniques has not changed but has now taken a secondary role in illustrating the basic concepts of decision making. Problem formulation and the analysis of results take precedence in this format over acquiring computational facility. Because of the direct relationship which exists between the computer and the increased usage of quantitative techniques, and because of its capacity for handling

routine calculation we have decided to incorporate its substantial use in the new edition of I.S.A. 385. The computer thus assists in the instruction of the general business student in this highly quantitative course.

Unfortunately, the physical limitations of the Univac 1100 in a University with over 16,000 students has restricted availability, at least in the near term.

This means that full implementation of our goals must await the installation of more modern equipment over the next two years.

IMPLEMENTATION

Because of the limited facilities and time, the topics of linear programming, PERT, forecasting and regression analysis were best suited to a computer aided instruction (CAI) approach [9].

As previously mentioned, the majority of students have been exposed to some of the computational aspects of linear programming in their prerequisite math courses. With CAI the emphasis could now be placed on formulation, suitability of the model, interpretation and sensitivity to data variability in a wide variety of applications. The main difficulty arose in finding a software package that was readily accessible to inexperienced users. Unfortunately, no L.P. package available to us met our criteria for ease of use and interpretation. This did not prove to be a problem as writing a program that required simple input and produced the desired output was not difficult. The resulting package has been tested in several I.S.A. 385 sections. As it turned out, less than one class period was needed to introduce usage of the keypunch and run a sample problem. Amazingly enough, little if any help was needed for any further assignments. Therefore our concern regarding computer orientation within the course proved unwarranted.

In addition to some routine problems, introduction to the dual and sensitivity analysis was greatly simplified. Students could compare the output and time for both the primal and dual. Also verification of the effects of changing the constants in the objective and constraints were readily understandable with the outputs in hand.

The application of CAI to PERT was also straightforward. Again a simplified software package was not available to us so a program had to be devised. With this program, the students had to develop their own code for input and the necessary calculations appeared in a chart along with a printout of the critical path. This left more class time for discussions on scheduling, setting dates which had

high probability of completion, computing expected loss in the cases where penalty fees were imposed and calculating costs of crashing. 223

As regression analysis and forecasting are the topics most requested by instructors in subsequent business courses, there has been much pressure to include them in this required course. This year we have attempted to streamline these topics by providing the students with an in-house handout covering an outline of the basic models from simple moving averages to multiple regression. The outline lists basic assumptions, advantages, disadvantages, interpretations and possible extensions of each model. A list of the control cards for the related computer packages along with sample outputs are included. In all, the outline is so self-contained that in a matter of four lectures one can cover material that would normally take close to half a semester.

Again the programs to calculate moving averages and exponential smoothing were written by us. We also employ SPSS for multiple regression. This package is so extensively used and widely available that we wanted students to become aware of its usefulness. This combination of computer programs frees the student from the computational aspects to concentrate on the visual presentations and relative importance of the various factors under consideration in a decision. (This has proved exceptionally valuable in subsequent courses where the Executive Decision Game was employed for practical experience in decision making) [7].

CONCLUSION

As this is our first semester of CAI in the introductory undergraduate quantitative methods course, tests to compare this approach with the more traditional offering have not yet been structured. However, on the basis of one instructor's test results a distinct improvement in quiz scores was detected in linear programming.

Two sections presently taking the CAI course were compared with two sections from the previous semester ($n_1=43$ and $n_2=62$). The tests were similar in content and covered the dual and sensitivity analysis in linear programming. A comparison of the mean scores showed the CAI classes scored higher (76.05 vs 68.66) with a significance level of .0465 [8]. Other classes did not maintain the same instructor for comparison.

In general, the department has been quite pleased with the initial results and will continue to implement them on the basis of what we have observed to be the following distinct advantages:

224 (1) The overall attitude of the students on topics for which CAI was attempted has improved observably [4]. Previously, text examples (particularly in linear programming) were so computationally involved that students would not complete or not be able to detect errors in the assignments. With CAI, students could focus on the more important aspects of the model. Consequently, many students were motivated to work on problems of interest in addition to those assigned.

(2) Homework problems which involved the use of the computer led to a more consistent format for solution presentation. This fact tended to reduce the amount of time necessary in grading. Also, it was felt that the amount of collaboration on assignments had been reduced. Computer printouts of student usage could also be checked for confirmation of each student's involvement.

(3) With CAI, more complex (and hence more instructive) examples could be addressed in homework assignments. This allowed the instructors freedom to include more of the qualitative and conceptual aspects of the course on exams. Again, this facilitated grading both in time and equanimity. Previous exams that were heavily computational were often hard to grade in a fair and consistent fashion if computational errors occurred.

(4) CAI allowed a more integrated teaching approach for the purpose of integrating several topics into a decision making framework. The consistency of this approach carried throughout appeared to provide the students with a firm basis and sense of direction at the beginning of each topic. This seemed to abate waning interest in the longer, more involved topics.

(5) CAI also freed the instructor from much computational redundancy, allowing lectures to become more motivational and conceptual in nature.

(6) The use of CAI has also structured the class to the extent that instructors in different sections may be able to cover the necessary topics comfortably within a limited time framework. This should allow for a more consistent foundation between sections upon which subsequent courses can build.

(7) The inclusion of forecasting and regression analysis has reduced the imminent need for the inclusion of a

second semester statistics course into the core.

(8) Finally, the use of streamlined personal packages used to solve particular problems gradually leads the student to the point at which he is ready to accept the more complex software packages available.

BIBLIOGRAPHY

- [1] Brabb, G. J.; and E. J. Livingston. "A Decision Analysis Approach to Business Statistics." Decision Sciences, Vol. 7, No. 3 (July, 1976) pp. 528-546.
- [2] Busch, Paul; D. T. Wilson; and L. J. Dolich. "Behavioral Objectives and the Quantitative Methods Course." Decision Sciences, Vol. 5, No. 1 (Jan. 1974), pp. 128-139.
- [3] Byrd, J. Jr.; and Patrick Espasita. "A Course Model for Teaching Concepts Related to the Successful Implementation of Quantitative Analyses." Decision Sciences, Vol. 7, No. 2 (April, 1976), pp. 366-376.
- [4] Cerveny, Robert P.; "Some Experiences with the Introduction of Computer Use into an Undergraduate Business School Environment," AIDS Proceedings 9th Annual Conference (Oct. 1977), pp. 120-122.
- [5] "Computer - Assisted Instruction: Many Efforts, Mixed Results." Science, Vol. 176, No. 4038 (June, 1972), pp. 1005-1006.
- [6] Cook, T. M.; and R. A. Russell. "Management Science: Present and Future." Introduction to Management Science, N.J.: Prentice Hall, 1977, pp. 433-453.
- [7] Henshaw R. C. Jr., and J. R. Jackson; The Executive Game, 3rd Ed.; Irwin, 1978.
- [8] Johnson, Robert R. "Statistical Inferences." Elementary Statistics, Mass.: Duxbury Press, 1976, pp. 371-380.
- [9] Tsia, S. W., and N. F. Pohl. "A Comparative Study of the Effects of Lecture and Computer-Aided Instruction on Student Achievement in Computer Programming Classes." Decision Sciences, Vol. 9, No. 2. (April, 1978), pp. 297-309.

THE PROJECTS WITH INDUSTRY APPROACH TO
TRAINING AND PLACING DISABLED STUDENTS
AS COMPUTER PROGRAMMERS

By

Robert J. Leneway

Michigan State Technical Institute and
Rehabilitation Center

"I have broken my bones at least a hundred times although I really can't remember," says Brian Short who at age nineteen is confined to a wheelchair with a congenital disorder osteogenesis imperfecta. Carl Dort, thirty-eight, from Mount Clemens, Michigan, is the father of three and was employed as an operating engineer with the Detroit Edison Company until a snowmobile accident left him paralyzed from the neck down. Carl and Brian are students of a new Computer Programming Training Project at the Michigan State Technical Institute and Rehabilitation Center near Plainwell, Michigan. This training project is the latest example of business initiated programs within the rehabilitation field. Initiated under the guidance of an IBM Program, there are now ten such projects in different states with another dozen or so in some stages of initiation or development.

The purpose of this paper is to share some special training and placing techniques used to interface the severely disabled with industry as used in the IBM Projects with Industry (P.W.I.) model that may also be usable in the "normal" data processing training environment.



At each I.B.M., P.W.I. project, the critical element has been the deep involvement of the local DP community in the form of a functioning advisory committee. The committee establishes prerequisites for admission, develops the curriculum, monitors instruction and progress, certifies the graduates as qualified entry-level programmers and assists in job placement.

With the high level of assurance of success, almost all graduates are placed in professional programmer positions and become successful programmers.

The P.W.I. Approach

This approach to placing persons with severe disabilities into competitively unsubsidized employment is called Projects With Industry (P.W.I.). It was first initiated by the Department of Health, Education and Welfare in response to the implementation of 1968 Amendments to the Vocational Rehabilitation Act.

Projects With Industry are among the most effective means of helping handicapped individuals into gainful employment. In the data processing industry, the IBM Corporation has been the leading catalyst in the initiation, implementation and development of new training PWI's for the handicapped. The ten existing handicapped training projects are located in a variety of educational environments including universities, Goodwill Industries, independent living centers and comprehensive vocational rehabilitation training centers. They are listed in the Appendix at the conclusion of this paper and currently include representation in the following states: Alabama, California, Connecticut, Maryland, Massachusetts, Michigan, Missouri, Ohio, Pennsylvania, and Virginia.

Each project in this PWI Program is based on the cooperative efforts of an interested rehabilitation agency, a competent and experienced training facility, a far-sighted business community, and qualified disabled students.

All of the existing projects follow the same PWI training model, but each responds to its problems in the way best suited to its environment.

Michigan's PWI

For example, the Michigan Project is located near Plainwell, Michigan, in the center of the Kalamazoo, Battle Creek, and Grand Rapids industrial triangle.

The State Technical Institute and Rehabilitation Center (STIRC) in cooperation with the Academic Computer Center at Western Michigan University serves as the rehabilitation training partner for the Michigan Project.

STIRC is a state-operated, full-service facility which offers a variety of vocational training opportunities. Severely disabled students are referred to the training through local vocational rehabilitation offices.

The computer programming class which meets daily from 8:00 A.M. to 3:30 P.M., is eleven months in duration. It is only part of the comprehensive rehabilitation effort at STIRC that also includes medical care, physical therapy, job readiness counseling, leisure time training and other services vital to the total rehabilitation process.

Business Advisory Council

The Michigan Project was started when representatives from the IBM Corporation contacted high level D.P. leaders from three communities and invited them to a meeting to help organize a computer programming training program for handicapped people.

Interested business advisors formed a Business Advisory Council (BAC) divided up into three standing committees and overall coordinating planning committee. The divisions of the sub-committees primarily follow geographic areas; and their roles are as follows:

The Admissions/Placement Sub-Committee screens prospective students, identifies organizations receptive to the hiring of the program's graduates, establishes key contacts within those organizations, utilizes appropriate committee resources during initial negotiations, and obtains information from employers as to the success and utility of graduates' training.

The Curriculum Sub-Committee assists in the development of the course curriculum, subjects it to periodic review and suggests modifications when appropriate.

The Business Education Sub-Committee arranges for educational resources external to the rehabilitation center, acquires literature, organizes site visits, and obtains guest lecturers and specialist information.

The membership includes representatives from both small and large data processing shops,

such as Amway, G.M., Michigan Bell, Steelcase, Foremost Insurance, Meijers, Rapistan, Brown Paper, IBM, Lear Siegler, Kellogg, Foremost-McKesson, State of Michigan, and Upjohn's.

The bulk of the Business Advisory Council consists of representatives of organizations which have made a continuing corporate commitment to this Michigan Project. Each organization may have several individuals nominated for active participation. Individuals not affiliated with an organization, who feel they can make a contribution to the success of the program, are encouraged to participate. Additional members are solicited whenever the committee and the director feel that the program will benefit from a new presence.

There are currently over thirty active involved business advisors. Non-business advisors include a physical, medical and rehabilitation physician from Mary Free Bed Hospital, a handicapped programmer and an occupational therapy instructor from Western Michigan University.

During the first semester of the project, BAC activities averaged over two weeks. BAC members volunteered their time and efforts in the following ways:

1. Screening prospective students.
2. Conducting analysis of tasks to be taught.
3. Hosting company visitations.
4. Conducting career orientation to data processing.
5. Conducting CAI activities.
6. Luncheons meeting with students.
7. Providing guest lecturers.
8. Making library donations.
9. Making DPMA presentations.
10. Surveying the architectural barriers of the member DP shops.
11. Recruiting new BAC members.
12. Conducting technical reviews of the students' progress.

During the second semester of this new three semester program, the BAC members have continued many of the activities started during the first semester as well as initiating new activities, such as:

13. Donating computer equipment.
14. Providing budget input.
15. Mediating conflicts with other training programs.
16. Presenting spinal cord seminars.
17. Designing student profile sheets.
18. Acting as employment assistants.

Employment Assistant Program

Each student has been assigned to a specific employment assistant selected by the BAC's Placement Committee. The primary objective of the assistant is to secure full time employment for the student. Responsibilities include assisting the student in preparing for employment interviews, assisting in arrangement for interviews, arranging special counseling or use of special facilities and services available

through STIRC and other appropriate work necessary to prepare the student for an employment interview.

A major responsibility of the assistant is the preinterview contact with prospective employers. The employment assistant is knowledgeable of a student's capabilities as well as any special changes in work stations, files, and terminals.

Employers are made aware of the student's capabilities and any special adaptations that may not be obvious during an interview.

Technical Reviews

In order to assure the employment assistant that the student is progressing toward the goal of competitive employment, the Curriculum Subcommittee conducts periodic personal reviews of the student's classwork. Four groups of reviewers working simultaneously interview each of the students for an hour, six times during their year of training.

Through the Review Board, students are able to determine whether or not their work is acceptable and relevant to what they will be doing once they are employed. The Review Board also provides students with feedback on their programming technique, logic, report layout, documentation, etc. For some, it may also present an opportunity for them to discuss their physical disability and whether or not it will limit them on the job.

Each student is given feedback indicating his/her strong points and weaknesses. When all of the reviews are finished, the instructor meets with the reviewers to discuss the individual students and the class as a whole.

After their first technical reviews, the reviewers often leave expressing amazement at what a severely handicapped individual can do when given the appropriate opportunities.

A.R.P.D.P.

The technical reviews by the Michigan Project grew out of information sharing sessions with other FWI projects. The newly formed Association of Rehabilitation Program in Data Processing (A.R.P.D.P.) organized to promote communication between and among programs designed to train disabled people in the area of data processing. Each of the ten existing projects are charter members of A.R.P.D.P. Each of the projects are tied into the Electronic Informational Exchange System through the local telnet system into the New Jersey Institute of Technology. This has helped to facilitate the sharing of information among the computer programming training FWI's. Future A.R.P.D.P. plans include developing systems for placement exchanges.

Summary

These are only a few of the ways in which a dedicated Project With Industry approach can be used to interface students with the industrial

community that they are being trained for.

The FWI approach to D.P. training can be both time consuming and difficult, but the benefits in the long run are well worth the effort. It provides an excellent approach to supplying receptive employers with proven competent students, which is what any D.P. Training Program should be striving for.

Data Processing For The Handicapped Projects

Birmingham, Alabama
Mr. John Gaston
Lakeshore Rehabilitation Facility
3800 Ridgeway Drive
Birmingham, AL 35209

Boston, Massachusetts
Mr. Henry J. Dufresne
Massachusetts Rehabilitation Committee
296 Boylston Street
Boston, MA 02116

Cincinnati, Ohio
Mr. Larry LeVine
University of Cincinnati
College of Business Administration
Cincinnati, OH 45221

Fishersville, Virginia
Mr. Kenneth Knorr
Woodrow Wilson Rehabilitation Center
Fishersville, VA 22939

Houston, Texas
Mr. J. Richard Young
Goodwill Industries
5200 Jensen Drive
P. O. Box 21185
Houston, TX 77026

New Haven, Connecticut
Mr. Malcolm Gill
Easter Seal-Goodwill Industries Rehab. Center
20 Brookside Avenue
New Haven, CT 06515

Philadelphia, Pennsylvania
Mr. William F. Nilsson, Jr.
Director of Handicapped Training
215 Franklin Building/16
3451 Walnut Street
Philadelphia, PA 19104

San Francisco/Oakland, California
Mr. John E. Velton
Department of Rehabilitation
1111 Jackson Street
Oakland, CA 94607

Plainwell, Michigan
Mr. Robert Leneway
State Technical Institute & Rehab. Center
Alber Drive
Plainwell, MI 49080

THERE ARE NO BLACKBOARDS

Ronald V. Carswell CDP
Texas State Technical Institute

In the Computer Science Technology (CST) program at Texas State Technical Institute-Waco campus, blackboards are not necessary. CST prepares entry-level computer programmers in 18 months by utilizing a systems approach. Statewide School-Industry Cooperative Committees specify entry-level skills by completing the task inventory, and from the Job Entry Specifications, CST faculty develop performance objectives. Advisory Committees annually evaluate the program's progress. Self-paced instruction permits students to progress at their own pace in a work-like environment and permits open-entry/open-exit, while mastering skills at the 90 percent level. Cooperative training is also available.

INTRODUCTION

The phrase, "There are no blackboards" signifies the uniqueness of the instructional program in Computer Science Technology (CST) at Texas State Technical Institute-Waco campus (TSTI-WACO). The phrase evolved during numerous prospective student visitations this past summer. From the center aisle of the learning laboratory, "There are no blackboards" was proclaimed to each prospective student to indicate that they might be participating in an instructional system which would vary greatly from their previous educational experiences. The phrase was soon adopted by our current students as a means of communicating that they were participating in this unique instructional program.

TSTI committed itself both philosophically and financially to the individualization of instruction. This means that ultimately, the pace of instruction will almost entirely be controlled by the learner. Ideally, the closer we get to this individualized, learner-controlled situation, the better will be instructional programs prepare students for careers.¹ What type of institution is making this bold commitment? TSTI is a State-supported institute offering one and two-year vocational/technical programs at four campuses located across the state.

INSTRUCTIONAL INNOVATIONS

The CST program prepares entry-level computer programmers. Entry-level is the skill level at which the

student must perform in order to graduate. Job Entry Specifications, a synonym for entry-level requirements, are a list of duty and task statements for computer programmers.² A systems approach insures graduates can perform the tasks required by the prospective employer. The School-Industry Cooperative Committee (SICC) assists in the determination of inventory skills by completing the Task Inventory Questionnaire. Advisory Committees assist in the final preparation of the Job Entry Specifications and annual review of the instructional program. The CST instructors prepare a precise set of performance objectives from the Job Entry Specifications before the development of self-paced instructional materials. Instruction is continually evaluated and analyzed to facilitate improvements.

School-Industry Cooperative Committee

Industry participation is encouraged by membership in the School-Industry Cooperative Committee (SICC). A School-Industry Cooperative Committee is organized to counsel the program and to make suggestions and recommendations for improvements in a specific program. The Committee's function is to provide closer cooperation and better understanding between industry, the State of Texas, and the Institute. A School-Industry Cooperative Committee provides a two-way system for communication between the Institute and the industrial community which is essential to all vocational/technical programs.³

The main function of the Committee

is the identification of program training needs. In order to keep training relevant, committees must take an active part in establishing an inventory of tasks the graduates may be expected to perform. The collective opinion of a large number of representatives is essential to establishing a comprehensive task inventory which meets the industry's needs for trained computer programmers.

Advisory Committee

The CST Advisory Committee members are selected from the larger membership of the SICC. The CST Advisory Committee members meet with the CST faculty and TSTI-WACO campus administration annually to review the operations of the CST program.

The Advisory Committee is charged with making recommendations to the CST program for establishing and maintaining current Job Entry Specifications. These should be in the form of a comprehensive task inventory.

The Advisory Committee can often help provide direct and indirect financial support by donating or securing donations of usable equipment, tools, and materials.

Committees take an active part in the evaluation of the instructional program. Valuable advice can be provided concerning computer room layout, the type and quantity of equipment needed to insure the program's facilities are adequate to meet the skill and knowledge requirements of industry.

The Advisory Committee functions in an advisory capacity to help insure the program is current with the needs of industry. The advice given by this committee can assist in producing graduates which are acceptable to industry and a valuable asset to the employer.

Task Inventory

The SICC members are surveyed periodically in order to complete the Task Inventory. A Task Inventory is a list of appropriate duty and task statements covering the tasks performed by workers in an occupational area. It may also contain identification and background information and may be used to collect occupational information from incumbent workers.⁴

The primary function of the SICC is to assist the CST program in establishing and maintaining a current Task Inventory in the form of a comprehensive questionnaire designed to identify the knowledge and skills required

for acceptable entry into industry as an entry-level computer programmer. The tabulated results of these questionnaires are used to assist the Advisory Committee and the CST program in developing Job Entry Specifications.

Performance Objectives

The instructors in the CST program translate the Job Entry Specifications established by the Advisory Committee into performance objectives. A performance objective simply describes what the learner will be able to do as an outcome of instruction.⁵ Specific objectives serve three purposes. For the instructional developer, they are valuable in planning and sequencing instructional activities. For other instructors, they indicate the probable outcomes for instruction. For the student, they describe exactly what the student must be capable of accomplishing, and provide him with a goal to be mastered.⁶

Objectives can be classified as terminal or enabling. Terminal objectives bear a close resemblance to the objectives specified by the Advisory Committee in the Job Entry Specifications. Enabling objectives are specified by the instructional program to bridge the gap between the student's previous skills and the specified terminal objectives.

Self-Paced Instruction

After the careful preparation of performance objectives, the instructor selects an instructional strategy. The selected instructional strategy prepares the student to demonstrate the achievement of the specified performance objectives whether they be terminal or enabling.

The self-paced instructional format utilizes an instructional package pertaining to one or more performance objectives. The approach is a learning experience which actively involves the student. The student is given the opportunity to conduct a self-paced study with the opportunity to review at will.⁷

A basic assumption made in the development of modules is that learning is a process which must be done by the learner.⁸ The generic term for a small unit of instruction which seems to be emerging in the literature is "module".⁹ These statements have important implications for the instructor preparing the self-paced instructional materials. The materials must be developed around sound principles of learning theory.

Different learning strategies are selected for the three basic instructional groups within the CST program. Programming-related courses require strategies which permit the development of concepts and terminology. Programming language courses require strategies which lead to the solution of problems containing specific instructional sequences. Application courses require the application of programming and programming-related skills to realistic problems.

The selected instructional strategy functions best in a work-like environment. We have borrowed generously from the work-related training methods of, for example, auto mechanics and welding. A work-like environment for the student consists of a work station which closely resembles the working environment of the computer programmer. The application project course requires the student to develop a subsystem complete with documentation. These projects are subsequently released to the cooperating agency.

Peer-tutoring is utilized to handle some learning situations. After careful explanation of a concept or the development of a particular skill with one student, this student may be called upon to share his skill with his classmates. Peer-tutoring provides a learning activity for the other student.

Self-paced instruction permits the student to master skills before continuing to other materials. A common level of mastery is the 90 percent level. Self-paced instruction also facilitates the simulation of a work-like environment.

Open-Entry/Open-Exit

The CST program is striving to implement an open-entry/open-exit system. The program accepts students four times annually. Upon completion of a major course, the student may start the next course. The program has adopted a position of "Never knowingly impeding the progress of the student". Students graduate as an entry-level computer programmer in 15 to 21 months. Progress is currently being made to provide an early exit for computer operators at 9 to 12 months. The operator trainee exit requires the substitution of an operator training course for a programming language. The normal TSTI registration processes are currently a limiting factor regarding the complete implementation of open entry. The student may start a CST course at any time the prerequisite courses have been completed. The open-entry/open-exit program will ultimately furnish graduates

to industry at numerous intervals throughout the year.

Coop Training

Industrial cooperative education is a cooperative effort between the student, school, and industry to produce a better trained student. Students may be given an opportunity to participate in one or two quarters of cooperative education in their field. A student may "co-op" after completing two quarters and prior to initiating the application project.

A cooperating training plan is developed jointly between the CST program and the employing firm. The plan indicates the specific areas in which the student will work and the expected outcomes of the student learning activities.

A CST faculty member visits the student in the cooperative training program each month. Written reports consist of weekly student progress reports by the student and performance appraisals by the employer.

Cooperative training provides the student an opportunity to work in his chosen field. By alternating work and study, theory and practice are more closely integrated and the student's motivation to study increases.

Cooperative training orients the student to the world of work. Because work experience brings the student into contact with co-workers from a variety of backgrounds, the student will have an excellent opportunity to develop greater understanding of other people. The student has opportunities for exploring their own abilities in relation to actual job performance.

Graduate Follow-up

A graduate follow-up program monitors the performance of graduates on the job. The graduate follow-up is completed after the critical six months period. Both graduate and employer indicate the training areas that should receive greater emphasis in the future. The task inventory and the graduate follow-up provide valuable contributions to overall effectiveness of the CST instructional program.

SUMMARY

The final test for any instructional program is the acceptance of graduates by industry. A systems approach maximizes the chances for acceptance of the graduate by industry. The early involvement by the SICC and Advisory Committee insures the validity of

the Job Entry Specifications. Careful preparation of performance objectives guide the development of meaningful instructional materials. A heavy emphasis on job-related programming assignments and a work-like environment should provide graduates with a smooth transition from school to job.

The instructional process is continuously monitored by a graduate follow-up program. The CST program relies very heavily on the advice and guidance of industry.

FOOTNOTES

1. Systematic Instructional Development Plan, Waco, Texas; Texas State Technical Institute, 1975.
2. Cox, William H., Inter-Office Correspondence to Dr. Clay G. Johnson on Terms and Standards, Nov. 5, 1975.
3. A working document on the roles of the School-Industry Cooperative Committees and Advisory Committees with foreword by Jack E. Tompkins, 1973.
4. Melching, William H. and Borchert, Sidney D., Procedures for Constructing and Using Task Inventory, The Center for Vocational and Technical Education, The Ohio State University, Columbus, Ohio, 1973. p.3.
5. Whelan, Donald W., Specification and Analysis of Performance Objectives, Waco, Texas; Texas State Technical Institute, 1977.
6. Russell, James D., Modular Instruction; Minneapolis, Minnesota; Burgess Publishing Company, 1974. p. 45.
7. Russell, p. 14.
8. Portlewait, S. M., Novak, J. and Murray, H. T., The Audio-Tutorial Approach to Learning; Minneapolis, Minnesota; Burgess Publishing, 1972. p. 1.
9. Russell, p. 3.

**EDUCATION ADMINISTRATIVE
APPLICATIONS**

THE COMPUTER BASED HOURLY CLASSROOM
ATTENDANCE SYSTEM IN
KALAMAZOO PUBLIC SCHOOLS

Jerry Fochtman
Kalamazoo Public Schools

In an effort to reduce absenteeism, increase student achievement and parental awareness, Kalamazoo Public Schools has developed and implemented a computer based attendance system. Individual classroom attendance information is automatically fed into the computer by each building's attendance office personnel. This information is then used to produce a variety of reports on student attendance patterns which are distributed to teachers, counselors and administrators. These reports are also utilized to comply with the state laws regarding pupil attendance. In addition, parents are provided with information regarding their student's attendance on a weekly basis.

Absenteeism among secondary school students has increased at an alarming rate over the past several years. Many school systems recognize this problem but haven't developed any programs to deal directly with student attendance. Kalamazoo Public Schools has developed and implemented a program which is based on the feedback of attendance information to parents, teachers, counselors and administrators.

Previous to September, 1977 students were allowed to miss up to 12 days per semester of classes before losing credit in a class. This was essentially viewed as "vacation" time by the students, and towards the end of a semester, our absentee rate would increase as students used up their "vacation" time.

Beginning in September, 1977 the attendance policy was changed to a zero (0) absence policy, and parents were to be notified each time a student missed any one class 3 times. For the average student who has 5 classes it could mean 5 teachers would have to contact the parent in some manner whenever the student was absent 3 times. It doesn't take long for one to see the duplication of effort that would be required in this type of procedure and the volume of records that would be needed. At this point it became apparent that in order for any attendance policy to work it needed feedback information on a massive scale in a short period of time. Kalamazoo Public Schools already had an extensive student data base on an in-house Hewlett-Packard 3000 computer. Student attendance information could be linked to the data base, condensed and reports produced which would relieve a great deal of clerical work.

DATA COLLECTION

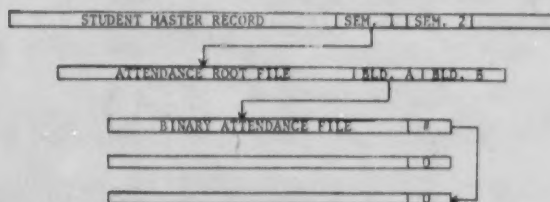
The biggest obstacle in any project like this is the collection of information in a form which could be used as input into a data base. Terminal

and keypunch input would require a great deal of person-hours; therefore, it would be costly and slow. The best medium for gathering the information would be an optically scanned form. Currently, Kalamazoo schools utilizes an OpScan-17 for processing a variety of tests and rating forms. This machine has the capability of processing attendance information, but it would be too slow. The cost of upgrading this equipment for faster turnaround would be close to \$1,000 a month. So the search started for a smaller scanner that could perform this task at a relatively inexpensive cost.

The J-M 5500 Data Scanner would read at the rate of one form per second, could be controlled by the computer, was easy to operate and inexpensive. Two of these scanners were ordered--one for each high school. They were connected to the I.T.T. Asciscope that was already present in each high school and are capable of sending information to the computer via telephone lines.

DATA STORAGE

A binary storage method was developed and linked to the student data base. The following diagram shows how the binary storage file was linked to the data base:



The student master record contains the record number for the individual student's entry in the attendance root file. This subfile points to the

236 first 128 word record in the binary attendance file. The field used as a pointer depends on the current building in which the student is enrolled. The first word in the binary attendance file always contains the record number of the attendance root file. The last word in the binary file, if not zero, points to the next record of attendance information for that particular student.

Each record in the binary attendance file contains 63 entries. One individual entry consists of two 16-bit words containing the following:

<u>WORD</u>	<u>BIT(S)</u>	<u>DESCRIPTION</u>
0	0-9	Course Code
	10-15	Section Code (First Part)
1	0	Section Code (Last Part)
	1-10	School Week Code
	11	Monday
	12	Tuesday
	13	Wednesday
	14	Thursday
	15	Friday

The course and section codes are the same as those used in scheduling student classes. They point to the description of the class that is contained in a master schedule of all classes. Information about the class such as title, hour, room and teacher is located in the master schedule file.

The week code points to an entry in a file that describes the actual calendar week which relates to the attendance information. The individual daily bits, if set to 1, indicate the student was absent that day during that school week.

ATTENDANCE SYSTEM OPERATION

A series of 14 computer programs are used to perform the entire attendance operation. The first program, PULSCHD, scans the data base for active students at specific buildings. A class look-up file is generated from the schedule information contained in the data base for these students. The FORMS program utilizes the class look-up file and produces the attendance scan forms sorted by classroom teacher. The printed forms are then delivered to the buildings for distribution to the teachers.

The completed scan forms are returned to the building's attendance office after the last class period of the school week. The attendance office personnel log-on the computer, place the attendance forms in the scanner and run the INPUT program. Not only does this program control the scanner, but it also checks the attendance codes for validity. If there is a problem, the INPUT program stops the scanner and informs the operator what type of error was detected and displays the last scan sheet that didn't have any errors. The operator must then locate the last successfully scanned form in the output hopper, pull the form that was scanned immediately after it and place that sheet back in the input hopper to be reread. Once this is done the operator signals the program to continue.

This procedure is followed until all the scan forms have successfully been read into the computer. The next step that the operator performs is to run the CHECK program which merges the scan sheet data file to the class look-up file that was used to generate the forms. The program informs the user of errors such as missing, can not match, or duplicate scan forms. If any of these errors occur, the operator runs the EDIT program to view and delete records from the scan forms data file. At this point it may be necessary to run the INPUT program again and append additional scan sheets to those already read into the data file. These steps are repeated until the CHECK program doesn't find any errors. At that time the user can instruct the computer program to initiate a batch job to complete the processing of the attendance information.

Some weeks classes are not held for certain periods due to assemblies or school might be closed an entire day because of vacation, weather, etc. When this occurs, the operator runs the CLEAN program which eliminates absences for specific class periods or entire school days. This is done to insure a student isn't marked absent for a period/day in which there was no school. The CLEAN program is run only after the CHECK program has merged the attendance scan file and the class look-up file. Upon completion of the program the user is again given the option to initiate a batch job to complete the processing of the attendance information.

The batch job executes the ADD program which updates the student data base with the attendance information. This program was designed to execute only when it could obtain exclusive access to the student data base. If any of the data base files are busy, the program will hyberate for fifteen minutes then check again.

Once the attendance information has been updated, the PULL program takes each student's class schedule and sums the number of absents for each class for all weeks except the current one. After all the previous attendance data is accumulated, the current week's information is added. During this process if the total absences for any one class adds up to be some multiple of 3, entries for this student are made in separate look-up files which will be used to generate the reports.

The REPORT program utilizes one of the look-up files and produces alphabetic listings or counselor listings showing each student's absences for all six classes during an entire semester. Listings for the homeroom teacher are produced by the HOUR program. A Child Accounting attendance information report is produced by the DAY program which also utilizes one of the look-up files from the PULL program.

The front and back sides of the post card that is mailed home to parents are produced by two separate programs named CARDS and ADDRESS. The postal permit information is also printed on the card to save the time needed to run the cards through a postage machine.

A section of code was added to the data base edit-

ing program EDIT to allow users to view and update attendance information. This also gives administrators and counselors the opportunity to examine individual student attendance in all classes over the academic year.

This entire process is then repeated the next week of school. Many positive reactions have developed because of the efforts to work with the attendance information generated by this application. Direct benefits include a small reduction in the absences rate coupled with a decrease in the number of student dropouts. More time is needed to examine the effectiveness of this attendance information system before reaching any conclusive results.

A DATA MANAGEMENT SYSTEM FOR A SMALL LIBERAL ARTS COLLEGE

Victor A. Meyer and Ignatia Frye

Marygrove College
Detroit, MI 48221

Abstract: Software development at Marygrove College for their Burroughs B-1700 system includes the Marygrove Data Management System (MGDMS) and the Marygrove Report Generation System (MGRGS). Each of these systems operate on the principle of small COBOL program modules performing specific functions for the purpose of efficient execution in a small main memory environment (96K bytes). Communication between these modules is accomplished through extensive use of queue files. After presenting an overview of the programming goals, a detailed analysis is made of MGDMS with specific attention paid to its three component parts - Master File Control (MFC), Master Terminal Control (MTC), and Office File Control (OFC). A brief description of MGRGS then follows with a look towards future software development at Marygrove.

I. Introduction

Marygrove College is a Catholic co-educational liberal arts college located in northwest Detroit with the established goals of promoting competence, fostering compassion, and encouraging commitment. The student body consists of approximately 900 students in degree programs (associates, bachelors, and masters) and more than 1600 in continuing education courses. In July, 1978, a new Burroughs B-1700 computer system was installed with the intent of handling all student and administrative programming needs. By mid-October of that year, the first phase of an internally developed data management system was put into operation. In January, 1979, the first phase of a report generation system (also developed internally) became operational. The report to follow is a brief description of these two systems which are called, respectively, the Marygrove Data Management System (MGDMS) and the Marygrove Report Generation System (MGRGS).

II. Description of System

The current computer hardware used at Marygrove College consists of a Burroughs B-1726 processor with 96K bytes of main memory and 4K bytes of high-speed control memory. Attached to the central processor are two dual-density diskpack drives providing over 174 million bytes of on-line data storage. Other peripherals include an 80-column card reader, 400 lpm train printer, supervisory console (SPO), and a multi-line controller to handle a communications network. At present, the communications network consists of three TD830 CRT-type terminals and a TC4000 printing terminal.

Burroughs software for the B-1700

consists of a Master Control Program II (MCPII) operating system, a Network Definition Language (NDL) compiler for setting up communication networks, a Command AND Edit (CANDE) message control system to facilitate program development through the terminals, and three language compilers - BASIC, COBOL, and FORTRAN.

Aiding the Director of the Computer Center is a staff consisting of a full-time programmer/analyst, a half-time administrative assistant, a part-time student programmer, and three to five part-time student clerks who are utilized primarily for data entry.

III. Major Programming Goals

Initially, a search was made of the various commercial vendors in an effort to find software that would conform to the administrative needs of Marygrove College. After examination, most of this software was rejected because it was either too expensive, would not run efficiently on the college's small system, or did not encompass the peculiar needs of a small liberal arts college. It became apparent that the only way to completely satisfy the needs of Marygrove would be by developing internally the necessary software.

In developing this software, COBOL was chosen as the primary language with four major programming goals identified as vital. First, it was established that programs must operate efficiently in the small main memory environment available. Second, some form of college-wide standardization was to be developed for the purpose of keeping to a minimum the amount of specialized programming performed. Third, the resulting programs were to be easy to operate with little to

no computer operating knowledge required. Finally, special attention was to be paid to error recovery capabilities to reduce computer down time resulting from system failure or operator error.

In fulfilling the requirements of the first programming goal, several techniques were developed to produce efficient processing in a small memory environment. Most of the programs were written as small modules performing only a few specialized tasks. While this process results in a more strung out operation, efficient processing is achieved by the smaller code files. Another technique for keeping down the amount of program code residing in main memory is to segment the code into areas of high, moderate, and low usage. Through this method, the programmer can specify the repetitious main procedure as high usage which keeps it in main memory and the initialization and termination procedures as low usage which allows them to be "rolled out" of main memory until they are needed. For handling those situations where the actions of one program module are determined by the actions of another, extensive use is made of queue files. As demonstrated in figure 1, a queue file is a technique for interprogram communication whereby one program writes information into an area of main memory and the operating system notifies a second program that information is ready to be retrieved from that same area.

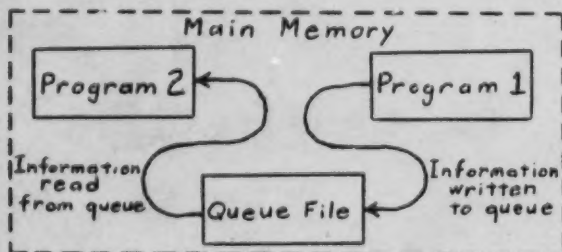


Figure 1.

Queue files to facilitate interprogram communication

In setting up a college-wide standardization of data as stated in the second major programming goal, a technique was developed which allows both the sharing and the separation of data that is used by the various administrative offices. Information involving personal names, addresses, and telephone numbers is maintained in a file called the Common Information File (CIF). By holding this information in common, any changes entered from one office will be reflected in all other offices that maintain information on the same person. In addition to the CIF, each office may have one or more Office Information Files (OIF) which contain the information needed by the office and which can be accessed only by the office for which they were created. In this way, privacy of information can be maintained. To manage CIF and the various OIFs, a Master File

Control (MFC) program was developed to provide the necessary linkage between the data records maintained on any given person (see fig. 2). A more detailed description of MFC and the rest of the Marygrove Data Management System will be presented in the next section.

The purpose of standardization is to cut down the amount of special programming required to satisfy the needs of the

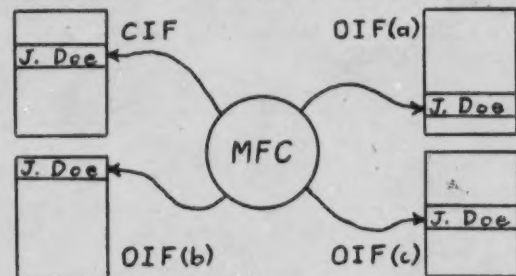


Figure 2.

All records on an individual are linked together by MFC

computer users. To achieve this goal, two methods of programming were established. The first method employs the use of COBOL library files containing the record structures of CIF and each OIF. By this technique, the same program modules can be applied to various situations simply by inserting the appropriate record structure into the program. The second method involves the concept of generalized skeletal program modules. As a specific application develops, only the patches to the skeletal program are written. The COBOL compiler then merges the patch file with the skeletal program file to produce a complete program satisfying the specific application.

The third programming goal involves ease of operation. A primary means of accomplishing this goal is through the development of control programs. A control program is any program which initiates, terminates, and monitors the activities of another program. Through the use of this type of program, an inexperienced operator needs only to know how to start and end execution of the control program. In those cases where additional operator control is required, English-like statements are accepted by the control program to cause various actions to occur.

The scourge of any computer center is down time caused by either hardware failure, misprogramming of software, or operator error. While little can be done to prevent hardware failure, much can be done in terms of automatic recovery to prevent extended delays when an error situation does occur. Whenever possible, programs are written so that error situations cause little more than an error message to be displayed at a terminal along with a request for the data in question to be re-entered. The most

240 severe error conditions are correctable either by re-initializing the program or terminating and restarting it.

IV. Marygrove Data Management System

The Marygrove Data Management System (MGDMS) serves the primary function of updating the Common Information File (CIF) and Office Information Files (OIF)

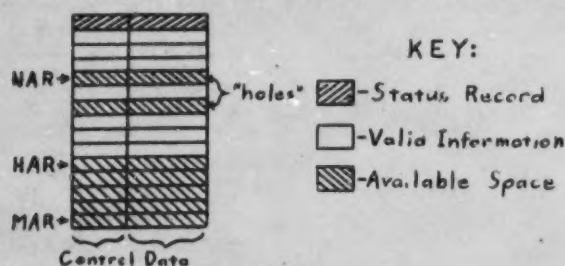


Figure 3.

Structure of MGDMS Data Files as a result of data changes on individuals whose information is currently stored in the system. Before examining MGDMS in general, an explanation of how the data files are structured is imperative.

CIF and all OIF's are structured along the same pattern (see fig. 3). Each file is divided into three parts: status record, valid information, and available records. The status record is always the first record of the file and contains the file pointers and their current status. There are three pointers maintained for each file. The Next Available Record (NAR) pointer contains the relative record number of the next record in the available record space to be utilized for holding valid information. The High Available Record (HAR) pointer points to the record one greater than the highest record in the file currently containing valid information. This pointer is used primarily for error recovery and to give some idea as to how much space is still available in the file. The third pointer is the Maximum Available Record (MAR) which always points to the last available record in the file.

The valid information part of a data file consists of all records which currently hold information that is accessible to the data management system. The available records part of a data file consists of all records which could be used to hold valid information as additional space is needed. Both the valid information and available record parts of a file are structured as double-linked lists. As new information is entered, data records are de-linked from the available records space and re-linked into the valid information part of a file. Conversely, as information is deleted from a file, data records are de-linked from the valid information and

re-linked back into the available record space of the file. This de-linking and re-linking is performed in such a way that the last record deleted from the valid information part becomes the first record to be added again when new information is entered. Hence, "holes" which develop in the actual physical location of the valid information part of a file are filled in first before the valid information part as a whole expands into the available space of a file.

In addition to the overall structure of the file, each record of each file is itself structured into two distinct parts - control information and data information. The control information part of a data record contains information regarding the status of the record, pointers for the double-linked list, identification of the person whose data is being stored on the record, and the last date for which there were any changes in the information or status of the record. The second, or data information, part of a data record contains the actual information being stored on an individual.

At present, CIF and each OIF is maintained by its own program module. The CIF is maintained by a program called Master Terminal Control (MTC) while each OIF is maintained by an Office File Control (OFC) program. The MTC program gets its name from the fact that it also serves the dual purpose of providing the

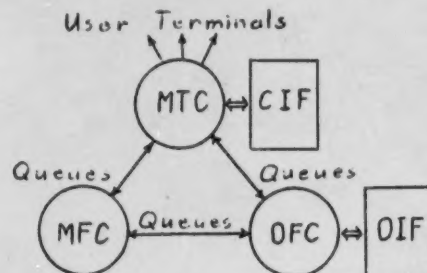


Figure 4.

Generalized Schematic of MGDMS necessary interface between MGDMS and users at computer terminals. Since there is no particular order by which information is stored in the data files, some type of search technique must be available to MTC and each OFC to determine where the data on a particular individual may be found. In keeping with the overall programming goals described earlier, this search operation was written into a separate module of its own called Master File Control (MFC). Hence, MFC provides the search techniques for all data files in MGDMS. Not only does this technique reduce the main memory requirements of MTC and the OFCs by eliminating the need for individual sort routines, but it also allows for centralized control of an individual's data across all administrative offices.

In general, MGDMS can be thought of in terms of the triangle of figure 4. The individual program modules (MFC, MTC, and OFC) perform specific tasks in support of the entire data management system. Interprogram communication between the modules is maintained through queue files as described earlier. When a user at a terminal requests information, MTC processes the request and determines whether it is a request for common information or office information. If it is a request for common information, then MTC transmits the identification number for the requested information to MFC which responds with the relative record number of that individual's data in CIF. MTC then retrieves the requested information from CIF and displays it at the user's terminal. If, however, the initial request was for office information, then the request is passed on to the appropriate OFC for processing. The OFC then submits a request to MFC for the relative record number of the individual in its own particular OIF. Once this information is obtained from MFC, the OFC retrieves the data from its OIF and passes it to MTC for display at the user's terminal.

In reality, the organization of MGDMS is more complex. Since each office has its own OIF and since each OIF requires its own OFC, then the typical operating mode of MGDMS consists of MFC, MTC, and several OFC program modules running and communicating with each other simultaneously. It is the job of MTC to spawn and terminate OFCs as they are needed by the various administrative offices. Hence, MTC becomes the controlling program for MGDMS by only allowing those OFCs which are currently active to remain in main memory. By this technique, virtually every office has access to its own information when needed, but main memory requirements are minimized by eliminating non-active OFCs.

A detailed illustration of MFC is provided in figure 5. As seen from the diagram, MFC maintains several queue files and disk files. All requests to MFC are submitted through the same Request Queue. In this way, requests from different program modules are processed on a first come, first served basis. Once a request is processed, MFC responds to the program module through one of several User Queues. Each module has its own User Queue which is assigned when a program module initially establishes communication, or "logs-on", with MFC. The Handshake Queue accomplishes this task of assigning queues by maintaining a list of all available User Queues. The computer operator can also submit requests and receive responses through the supervisory console (SPO). In addition to these queue files, MFC also maintains four disk files (Search, Link, Journal, and Info) to support its four major functions.

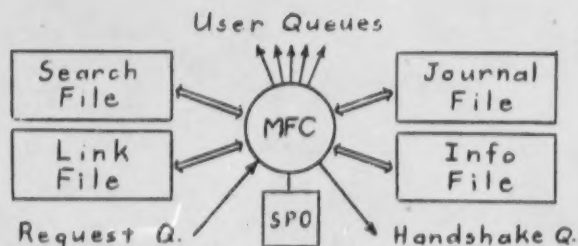


Figure 5.
Detailed Diagram of MFC

The first major function of MFC is to provide the necessary search and link mechanism among all data files. To accomplish this goal, MFC maintains two disk files called Search and Link. When presented with an identification number through the Request Queue, MFC uses the number as the actual relative record number of a pointer in the Search File. This pointer gives the relative record number of the first of a group of records on the individual in the Link File. This group of records in the Link File constitutes a double-linked list containing the relative record numbers of all of the individual's data in CIF and any OIF. The first record in the list always contains the pointer to the individual's common information. One or more pointers to various office information is then linked to this first record - thus providing centralized control over all data being stored on an individual.

The second major function of MFC is to provide security against unauthorized use of data. To accomplish this task, MFC stores in the Info File a list of security codes - one for each office incorporated into MGDMS. Before a module is allowed to log into MFC, it must present to MFC the correct security code assigned to it. In addition to this initial check, a log entry is made into the Journal File of all security violations.

The third major function of MFC is to maintain a log of all activities involving data changes in MGDMS to allow for recovery against loss of data due to some catastrophic event. To this end, MFC generates a new log record in the Journal File each time a data record is created, deleted, or modified. This journal record contains information such as date and time of the change, the terminal from which the change was made, the office which made the change, and the identification number of the data involved. This Journal File is maintained on a diskpack separate from the main data bank. It is retained until the main data bank is backed-up safely on other diskpacks.

The final major function of MFC is to provide billing information on the extent to which each administrative office has used MGDMS. This information is also logged into the Journal File for later analysis.

The next program module to be discussed in detail is the Master Terminal Control, or MTC, which is illustrated in figure 6. This program module serves as the controlling program for the entire data management system and, hence, has the

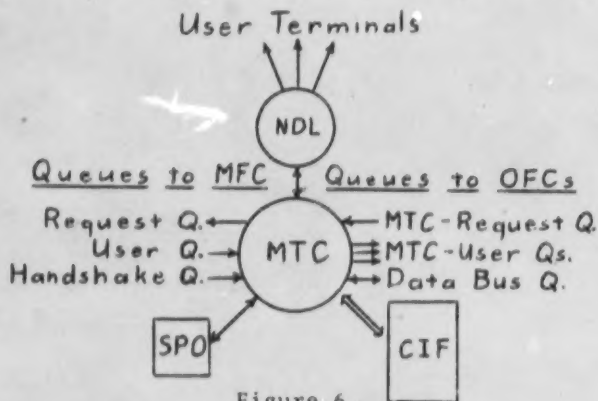


Figure 6.

Detailed Diagram of MTC

most complicated queue structure. First, MTC communicates to MFC through the Request Queue, and a specific User Queue as described earlier. To communicate with the various OFCs, MTC maintains another set of queues consisting of an MTC-Request Queue and several MTC-User Queues which are similar in nature to their counterparts for MFC. Since actual data must also be passed between MTC and the various OFCs, a two way Data Bus Queue is maintained and controlled by MTC. For communication to individual terminals, MTC requires still another queue to a program module written in Burroughs Network Definition Language (NDL). This NDL module describes the configuration of the data communication network and provides the necessary linkage between terminals and programs. Similar to MFC, the computer operator also has control over the operations of MTC through the supervisory console, or SPO. In addition to these files, MTC also maintains one disk file - the Common Information File, or CIF.

There are three major functions performed by MTC. The first major function is to provide the necessary interface between the user and the data management system. To accomplish this function, MTC generates video forms to be displayed at the terminals to provide an easy guide to the data management user. Through the use of these forms, people with little or no previous computer experience can be trained in minutes to perform data entry.

The second major function of MTC is to maintain CIF. To this end, MTC must keep track of file pointers and maintain proper linkage between the areas of valid information and available record space as described earlier.

The third major function of MTC is to control the spawning and termination of OFC programs as well as the routing of

messages and data to and from each OFC. When a person from a particular office first logs into MGDMS, MTC checks to see if that office's OFC program is currently active. If the program is active, then MTC routes all communications and data through that same running copy of the program - thus preventing the spawning of a second copy of the same program. If, however, there is no current running copy of the required OFC, then MTC will zip the necessary control information to the operating system to start up a copy of the OFC program. The net result of this technique is that several people from the same office can be working simultaneously without requiring the need for additional main memory space.

Another feature of MTC allows any administrative office to use any terminal. Hence, no card backup system is required in case of equipment failure. If a terminal in use by any one office breaks down, then that office merely has to switch over to any other available terminal to continue their work. When a user at a terminal indicates that he or she is finished with their work, MTC automatically sets up the appropriate messages to

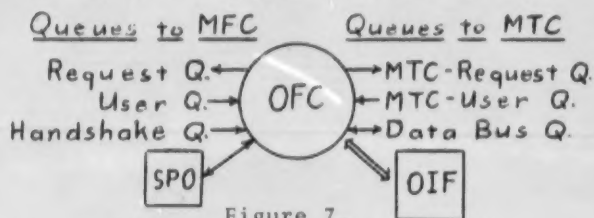


Figure 7.

Detailed Diagram of OFC

bring the executing OFC program to a normal termination.

The final program module to be discussed in connection with MGDMS is illustrated in figure 7 and is called the Office File Control program, or OFC. Like MFC and MTC, the OFC also has queue files for communicating between other modules in the data management system. One set of queue files allows for communication and transfer of data between OFC and MTC. A second set of queue files allows for communication between OFC and MFC. Also like MFC and MTC, OFC allows for direct communication to and from SPO. The only major function performed by an OFC program is to maintain the Office Information File, or OIF, for a particular administrative office. To accomplish this function, OFC maintains and updates pointers and links as described earlier. OFC also performs the necessary formatting of the data into a video display form for transmission to the user terminals.

Since each OFC program performs the sole function of managing an OIF, programming new OFC modules becomes a very easy task - usually requiring no more than a day or two. The only programming is the file format and a video form for

presenting the data at a user terminal. The entire procedure section of the program is exactly the same for all OFCs. Hence, the use of a skeletal framework for OFCs is utilized with only the patches actually being programmed.

V. Marygrove Report Generation System

While the Marygrove Data Management System primarily serves the function of maintaining the information data bank used by the administrative offices, another system, called the Marygrove Report Generation System (MGRGS), utilizes the data bank to generate the necessary reports for the college in general. Most of these reports are processed by extracting only that information from the data bank which satisfies certain criteria and then reporting that information on paper in a specific format. To facilitate the production of these kinds of reports, the MGRGS consists of a group of program modules which are sequentially executed to perform the required tasks.

The entire process centers around a special type of disk file called a List File. This List File is composed of the names, identification numbers, sort keys, and pointers to data records that have other needed information on each of the individuals involved. Besides the identification information, other information retained for sorting purposes include zipcodes and two general purpose sort fields that are user defined. Data record pointers contained on the List File include pointers to the individual's CIF record and up to three of the individual's OIF records.

The general procedure for generating a report is outlined in figure 8 and consists of three distinct phases. To facilitate the first phase (or Extraction Phase) of MGRGS, the concepts of COBOL

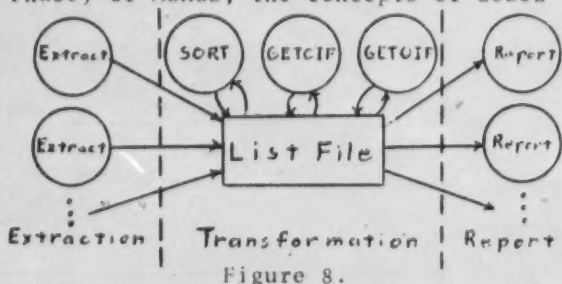


Figure 8.

Schematic of Marygrove Report Generation System (MGRGS)

library files and skeletal programs are used to extract information from the data files for the purpose of building a List File. In creating an extract program, the programmer needs only to indicate which library file contains the appropriate file structure and to program the necessary patches which set down the criteria by which records are to be selected for inclusion in the List File. The process is usually completed in minutes.

After the initial extract program is executed to create a List File, additional extract programs can be used to access other data files to expand the List File.

In the second phase, or Transformation Phase, a group of complete program modules operate on the List File to prepare it for use by the actual report programs. Examples of these types of programs include a SORT program module which will sort the List File along any one or more of the fields defined within the List File. Two other programs, GETCIF and GETOIF, interface with MFC to add pointers and information from CIF and any OIF to the List File.

The final phase is called the Report Phase and performs the actual generation of administrative reports. Certain reports, such as simple listings and mailing labels, are standard for any operation and consist of complete program modules. Many reports, however, must be written from scratch to handle specific types of needs. Since many of these types of programs cannot use program skeletons, the idea is to keep down programming time by letting the rest of MGRGS perform most of the work. In most cases, the programming of a report program should only involve accessing the contents of the List File to obtain pointers to the main data bank where the necessary information is obtained to produce the desired report.

V. Future Plans

At present, MGRGS does not have a controlling program which will automatically spawn and terminate program modules as needed in the report generation process. The next phase of software development at Marygrove, however, does include the programming of such a controlling program. The idea is to establish a small report generating language that can easily be learned by any part-time computer clerk. This language would consist of simple English statements which identify the order in which programs modules are to be executed and which sets the necessary parameters for these modules to perform their appropriate tasks. Once the controlling program is started, it will read the report language program and automatically spawn and terminate the modules necessary to perform the desired actions provided there are no errors. If errors are encountered, then the nature of the error will be reported back to the student clerk for correction and re-starting from the point of the error.

While MGDMS and MGRGS can in no way compare with the full powers of a complete data base system, they are sufficient to handle at least Marygrove College's current and foreseeable future needs in terms of computing power. It is hoped that these systems will do their part in helping Marygrove obtain its stated goals as outlined in the first paragraph.

DATA PROCESSING IN
THE YEAR ROUND 45/15
PLAN SCHOOL DISTRICT

Donald J. Vinlove
Valley View School District
Romeoville, Illinois 60441

This paper is a presentation of the unique needs of the year round school, and the use of data processing to satisfy these needs. The presentation is made to enable both the subject school district and other school districts to better understand and become aware of data processing in the educational setting. Users addressed and concerned with this topic are administrators, coordinators, and others involved in delivery of data processing services to elementary and secondary school users. The presentation is of significance to not only year round schools, but to the general awareness and appreciation of data processing in education.

The School District

Valley View School District 365-U is located in Will County, Illinois, 35 miles southwest of Chicago. The school district covers 38 square miles in which are located two cities, Bolingbrook and Romeoville. Student population of the school district is about 13,800 students with 11,000 dwellings. The student population is housed in fifteen buildings which consist of two high schools, four middle schools, and nine elementary schools. The central or district office is located in the lower level of a public library building in the city of Romeoville. The 45/15 Year Round School Plan was implemented in the Valley View School District in 1970 in the then elementary school district. In 1972, the school district and an annexed high school became a unified school district. The Year Round School Program encompasses grades K-12. A short explanation of the 45/15 Plan follows.

45/15 Plan

Pupils are divided into four groups, A, B, C, and D, around each of the elementary school buildings. Children in the same family or same neighborhood are placed in the same attendance group. Legal holidays, Saturdays, Sundays, vacation periods at Christmas, Easter, and the Fourth of July have been designated as times when the schools are closed for all classes. Each of the four groups of pupils attends approximately forty-five class days, and then each group has a fifteen class day vacation, hence the 45/15 terminology. Attendance patterns of each of the four groups are staggered so that only three

groups attend classes on any given day. The four groups remain in the same order of rotation. In a calendar year, each group is scheduled for 180 class days; that is four sessions of 45 days each. Classrooms and buses are scheduled for use on a year round basis. Teachers have the opportunity to work on an individualized contract.

Hardware

The initial computer hardware, installed in 1973, was an IBM System 3 Model 10. In 1976, the System 3 was upgraded to a Model 12 cardless disk system. The system uses batch input via floppy disks prepared on four IBM 3741 data entry machines. The central processor is an IBM 5412 with 64K of main memory. An IBM 3340 disk drive is used with seven 70MB IBM 3348 data module disk packs. The single printer is an IBM 1403 465LPM unit. Looking to the future, the school district anticipates delivery of an IBM System 38 in early 1980.

Software

Software used with the current hardware consists of a municipal budgeting and accounting system, "MBA", obtained from the State of Washington. The system has been customized to a great extent for use in the school district. The district purchased, altered, and customized an IBM student scheduling package to work with the 45/15 plan. The remaining various applications offered by the school district are custom applications and have been developed over the years by the

local staff. The system uses RPGII programming language and OCL source language.

Staffing

Current data processing staff consists of director of data processing, operations manager, programmer/operator, and operator/clerk. Most of the batch data entry activity is done by the user departments.

"Outside User"

The school district provides computer services to the Village of Romeoville (110 employees) and the Bolingbrook Park District (60 employees). Both of these entities receive the municipal budgeting and accounting package and the current payroll package resident in the system.

Present Applications

Applications currently utilized by the school district are:

- accounting, using the MBA package.
- accounts payable.
- purchasing is a customized system in which requisitions are prepared manually. Purchase orders are computer produced with vendor checks following through the system.
- a central warehouse application includes an inventory requisition system.
- Human Resource Information System, "HRIS", is the name given to the personnel system designed by the Valley View staff in 1975.
- student records is another application. Within student records is included a central enrollment facility with census capability, student scheduling, and grade reporting for secondary schools on a quarterly basis, and on an annual basis for elementary schools. A special education sub-system for tracking special education students through the school district system is contained within student records. An attendance sub-system is contained within student records which aids the district in preparing the annual state attendance report. A skeletal transportation sub-system is also contained in student records, although the district utilizes the services of ECOTRAN Incorporated for bus routing. The district owns its buses with a sub-system currently being developed for vehicle maintenance and parts inventory.
- Another partial application is in the instructional area in which office education courses utilize the data entry 3741 machines to train data entry personnel. Computer concepts courses are also offered in the sec-

ondary schools. Additional instructional systems are being considered for long term development.

245

Unique 45/15 Considerations

Unique year round 45/15 considerations for data processing are as follows. In the personnel system, individual teacher contracts are prepared annually. These contracts contain a per diem rate for each teacher determined by their position on the salary schedule. These individual contracts are initially printed in a calendar format and distributed to each staff member. These staff members in turn indicate days they intend to work the following fiscal year. The contracts or calendars are then returned to the personnel department, where they are entered on the data entry machines and submitted to data processing. Formal calendar format contracts are then printed and distributed to each employee. Personnel also handles a daily teacher substitute list utilizing a substitute bank. These substitutes are assigned by track and day breaks. Unique consideration in a year round school system is staff coverage for holiday periods during which many year round teachers may schedule time off.

In the payroll area, we utilize a true twenty-six pay cycle for all employees. The district currently employs approximately 1,800 people. The 1,800 people consist of regularly salaried employees, hourly or unit paid employees, and daily paid people. The normal deductions, such as teacher retirement, Illinois Municipal Retirement Fund deductions, credit union deductions, savings bonds, tax sheltered annuities, and so on, are allowed for in the system. There are per diem rate breaks maintained at 10, 99, and 170 days for the teaching staff. As our part-time or substitute employees reach these day breaks, retroactive pay is made from day 1 adjusted to their new daily rate. Normal full time teaching contracts are from 180 to 230 days. Payroll maintains a file of these contracts and makes per diem rate changes as necessary.

In the student record area, student scheduling is for a true four quarter year, running from July 1 through June 30 with course masters built accordingly. The senior high schools close out each semester or two quarter period, and schedule for a full year. Track considerations must be given when building the course master schedule as students must be tracked in courses corresponding to their attendance pattern. Courses may be multiple track, such as A, B, C, or merely single track. The student scheduler takes this into consideration and slots only applicable students into correct courses. Mid-year graduates are allowed at each

Attendance difficulties sometimes arise because of the on-off track set up of student attendance, with off track students influencing those students currently attending. Student grade reporting or mark reporting is by track year round. Certain considerations must be given to tracks that may be terminating in the middle of a grade reporting quarter as teachers, staff, and students will be off track or finishing a track three weeks prior to the end of a quarter. Therefore, grades must be assigned before these teachers go "off track". Students are graded for the entire quarter and receive their report cards at the normal end of the quarter, irrespective of when their track ends.

In the accounting area, there is normal quarter and fiscal year reporting. Also, there is the calendar year end reporting. Special consideration and planning must be given since there is no off or slack time which a traditional school district normally experiences in late June or July. These reports must be prepared and forwarded as part of the normal working year.

Conclusion

In conclusion, it may be said that the speed and efficiency data processing provides allows the year round school to function efficiently with no off time or year break. The Valley View systems and procedures are not as unique as, perhaps, they are a more efficient way to run a school district. The entire data processing effort is undertaken with a budget impact of less than one half of one percent of the district operating budget.

A DEGREE PROGRESS AUDIT OF STUDENTS BY COMPUTER

James E. Greene, Jr.
Lester A. Singletary
Georgia State University

Georgia State University (GSU) uses a computerized degree audit system to routinely inform students of their progress in fulfilling degree requirements. The degree audit results from comparing a student's course history (including transfer courses) to a specified set of degree requirements referred to as a "mask." Only one "mask" need be developed for each degree program. The system permits an individualized program of study for a student and the modification, as needed, of the general "mask" for a student. The system has proved effective in providing advisement information to a large number of students enrolled in a great variety of degree programs.

Introduction

It is often said that necessity is the mother of invention. The development of a computerized degree-auditing system to track the academic progress of students at Georgia State University, a large urban institution, was probably inevitable in view of several factors which define the character and operation of the institution. The present system called "Programmed Academic Curriculum Evaluation" but better known by its acronym, PACE, is an outgrowth of a manual system which was used to keep track of the progress of fewer than 4,000 students toward one of the two degrees then offered. It takes little imagination to appreciate that the present task of keeping up with well over 33,000 active students in 42 different degree programs and 183 majors and concentrations dispersed throughout six degree-granting colleges requires a comprehensive yet flexible approach. The Programmed Academic Curriculum Evaluation (PACE) approach fills this need and provides a wealth of valuable information for program planning by students, academic advisement by faculty and counseling personnel, planning of class schedule offerings by academic deans and department heads, degree eligibility auditing by the graduation office and a variety of other applications essential to the academic business of an educational institution. There are many situations in which it is necessary or useful to know the specific courses a student has completed toward his or her degree and the remaining requirements which must be satisfied before the student may be graduated.

To further place the PACE system in perspective and context, it may be useful to indicate that the University, like many others, has undergone a period of very rapid growth and change since gaining its autonomy as an independent entity within the State University System of Georgia in 1955. The university campus consisting of eleven major structures, most of which were built or acquired over a ten-year period at a cost of \$68 million, is located 2½ blocks from Five Points, the center of the financial and commercial district of Atlanta, Georgia. The University sees its role as meeting the need for a broad range of educational opportunities in the largest population center in the state. The University is a non-residential commuters' school with an enrollment of some 20,000 students. Over 75% of GSU's students work (most full-time) and the average student age is 27. Some 46% of the student body is enrolled in night classes. Females represent over 54% of the enrollment and almost 7,000 students are enrolled in graduate programs. There are currently 844 full-time faculty members. The rapid growth of enrollment and the proliferation of degrees, majors, and specialties have created a variety of educational and operational problems which have been addressed, in part, by the development of a computerized tracking system to monitor student progress and achievement.

Utilization of the PACE System

The PACE degree audit system has become a very valuable tool which is extensively used to facilitate the academic process. With a constituency composed of

248 older, more mature students, a significant amount of program planning and academic scheduling can be undertaken by the students themselves and student counseling resources can be shifted to aid students who wish to have additional assistance or who may be experiencing particular problems. The computer-generated output, sometimes called the program of study evaluation form (See Appendix A), is produced each quarter and distributed to the student's academic college. In some cases, the college may mail the evaluation directly to the student, together with information concerning arrangements for any required advisement, a registration permit and appointment notice, the schedule of classes to be offered from which the student may choose, and other instructions. The student may then review his or her own progress, review remaining requirements to be completed, and make informed selections concerning the courses to be taken during the next quarter. In some cases, depending on the nature of the student's program and other circumstances, the student may be instructed to bring the evaluation form to meet with a faculty member to plan future courses to be taken. Copies of the PACE evaluation can be prepared as needed for selected individual students who may have applied for graduation or need other specialized advisement or counseling services. Reports can also be prepared as needed for a variety of other legitimate academic and administrative uses.

Design, Structure, and Concepts of the System

The purpose of the PACE degree audit system is to inform a student of his progress in meeting the requirements for his chosen degree program in terms of which requirements have been satisfied and which requirements are outstanding. To accomplish this objective it is essential to first define the degree requirements and then to have a means of comparing these requirements to the courses a student has taken. This would be quite a simple task if the requirements were defined in terms of an exact number of predetermined courses in which there were no exceptions. That is, if the degree program called for courses A, B, C, and D, and only these courses, then as a student took courses A, B, C, and D it would be a simple matter to record and inform the student of his progress. Unfortunately, degree programs are rarely defined in such a simplistic and rigid manner.

The key to GSU's degree audit system is the notion of a "requirement." A degree program is defined in terms of a set of requirements (known as a "mask"),

which are organized into various areas such as humanities, mathematics, electives, etc. A requirement for a mask can be specified in a number of ways which include:

1. A specific course or courses
2. A selection of one or more courses from a given list
3. A specified number of hours from a given department and at a specified level (i.e. Freshman, Senior, etc.)
4. A specified sequence of courses (a given set) or a choice of a set from a given number of sets

As one can see, a requirement is not necessarily a particular course.

The requirements for each degree program are given in the University's General Catalog. It was desirable at GSU to specify degree requirements for the computer as closely as possible to the form in which they were printed in the catalog. This meant writing the requirements in English so that almost anyone could understand how to tell the computer what the requirements were for a degree program. Consequently, a computer program was written to analyze degree requirements written in English phrases according to a simple set of rules and to use this information to determine from a student's course history his progress in a particular degree program. The following are three examples of typical requirements and how they would be written for the computer:

1. ENG 111
2. (ENG 201 and ENG 202) or (ENG 204 and ENG 205)
3. 10 hours from the Dept. of ENG

A mask would contain degree requirements written in this manner.

GSU has both rigid degree programs as well as very flexible ones. Then there are degree programs that vary somewhat for each student. There are several hundred students in some of the 42 different degrees and 183 different majors and yet only a few in other specializations. GSU's degree audit system can handle all of these variations. Only one set of requirements need be developed for a given degree program regardless of the number of students in the program. If necessary, the general degree requirements can be modified for a student by updating the student's record with individual course requirements which can be substituted for normal requirements. The program of study for a particular student can be modified if desired to reflect additional requirements for that individual student.

As previously stated, degree requirements are organized into areas. Each area is subdivided into two parts: A) Those courses which have been taken to satisfy requirements in the areas, and B) remaining requirements to be satisfied. As requirements are satisfied they are omitted from the student's next degree audit. Courses which are not applicable to the student's degree program are listed at the bottom of the degree audit form and are designated as such.

Evaluation of Various Credits

For the degree audit system to be really effective, it had to be capable of handling all types of credit. This includes credit earned at GSU, acceptable transfer credit and credit by examination. Obviously, transfer credits from other institutions generally do not follow the same system of course designations. Although credit from another college may be transferrable to GSU, all the credit may not be applicable to all degree programs at GSU.

A system was implemented to classify transfer credit in a standard manner regardless of the originating institution, and to permit each academic department to indicate which of the transfer credits would be acceptable for a given degree program. A set of Universal Course Designators (2-4 characters) based on the HEGIS academic areas was developed. A course number is assigned such that the first digit is 0 (which indicates that the course was transferred), the second digit indicates level (i.e. 1 for Freshman, 2 for Sophomore, etc.) and the last digit is a sequence number. GSU does not have an Engineering School. So, three Senior-level transfer engineering courses from another college would be designated ENGR 041, ENGR 042, and ENGR 043. If there was also a sophomore engineering course transferred it would be designated as ENGR 021. Credit by examination is treated as transfer credit. The grade assigned indicates that the credit was obtained by examination (i.e. CR = Credit, DX = Departmental Exam, etc.)

At the time a transfer student is accepted, an "Evaluation of Transfer Credit" form is prepared by the Admissions Office using the system of classifying transfer courses. Two copies of this evaluation are forwarded to the academic department in which the student is accepted. The academic department then determines which transfer credits are acceptable by indicating beside each transfer course on the evaluation how each transfer course will be used in the designated degree program. This is done by equating each transfer course

with an equivalent GSU course. If the course is not acceptable, the Department simply omits equating the transfer course. Then all transfer credit is entered into the student's computerized record along with the GSU course to which it has been equated.

Computer System Used to Support The Degree Audit System

The PACE degree audit system is programmed in COBOL and is run on a Univac 70/7 computer. The degree audit system does not directly use any of the Student Record System files. Instead, an interface file containing the necessary data is extracted from the Student Record System and fed to the degree audit system. This adds a tremendous amount of flexibility for both the Student Record System and the degree audit system. Therefore, each system can be independently modified and maintained without having any adverse effects on each other. An on-line computer system is used by the academic colleges to update and maintain their degree requirements (mask) in a machine-readable form via computer terminals.

Additional Options

In the space available, it is impossible to fully describe GSU's degree audit system and all of the many options that the system offers. However, the more important additional options are as follows:

1. Required exam information is printed on the audit.
2. A minimum grade can be specified for a requirement.
3. The audit is formatted such that the student's address is in a location that will permit use of a standard window envelope.
4. The system can be directed to use or not to use a course which has been taken more than once for credit.
5. The requirement as printed on the audit can be phrased in any manner desired and is not limited to the rules for specifying requirements.
6. The user specifies on which side of the page the requirements for an area are to be printed.
7. Transfer credits can be "forced" to satisfy a particular requirement without having to equate the transfer course to a GSU course.
8. Different individual messages or comments can be printed on a student's degree audit.

9. Three standard lines of "Heading" comments and two lines of "Footnotes" can be specified for each degree program.
10. The advisor's name can be printed on the audit form.

Ironically, the biggest disadvantage of GSU's degree audit system may be its tremendous flexibility. On occasion, users attempt to define their degree programs for maximum flexibility so that the system will automatically handle the few exceptions that occur. Of course, attempting to handle all possibilities can easily and unnecessarily complicate any system. The goal should be a balance between maximum flexibility and optimum efficiency. Finally, the system obviously requires accurate student records and accurate definitions of degree requirements in order to produce accurate and usable degree audits. This includes updating a student's record with individual exceptions that might be made for a student by an advisor or other appropriate official. However, if all records are accurately maintained, students can be easily and accurately informed of their progress in fulfilling the requirements for their chosen degree program. This is particularly important as students near graduation.

Summary

The experience at GSU with the PACE degree audit system has been positive and beneficial. It has been demonstrated that this approach to academic performance monitoring can be effectively applied in situations involving a diverse student population who are pursuing a variety of heterogeneous programs and specialities. The strong points of this system include its simplicity, flexibility, and high degree of user control of output formats. The system has become a key element in the academic advisement process for students, faculty advisors, student services personnel and academic administrators. The concept of the system is simple and straightforward -- it evaluates a student's degree progress to determine the requirements which a given student has satisfied and identifies any requirements that remain to be completed. The computer examines the student's course history based on directions contained in the form of an evaluation "mask." It is believed that this approach to degree auditing has wide application in a college or university setting and could be adequately supported by a variety of computer installations.

SAMPLE DEGREE AUDIT

GEORGIA STATE UNIVERSITY
COLLEGE OF ARTS AND SCIENCES
UNDERGRADUATE DIVISION

ADVISEMENT EVALUATION
AS OF MAR 20, 1979
FIRST QUARTER - FA 76

STUDENT JOHN AVERAGE
123-45-6789 355-9999
DEG: BS CH CAT ED-7

THIS EVALUATION IS PROVIDED FOR ADVISEMENT. IT IS NOT AN OFFICIAL RECORD.
PLEASE REPORT ANY CORRECTIONS TO ROOM 725-G OR PHONE 658-2291.
ALL STUDENTS MUST MEET THE RESIDENCE REQ. AS OUTLINED IN THE UNIV. CATALOG.

1. HUMANITIES (20 HRS)

A. COURSES TAKEN IN THIS AREA

ENG 111 P 5.0 TRN
ENG 211 A 5.0 TRN
ENG 299 DX 5.0 TRN
DRAM 104 A 5.0 FA77

B. REMAINING REQUIREMENTS IN AREA

2. MATH & NATURAL SCIENCES (20 HRS)

A. COURSES TAKEN IN THIS AREA

MATH 105 A 5.0 FA78
GEOL 101 A 5.0 TRN
GEOL 102 A 5.0 TRN

B. REMAINING REQUIREMENTS IN AREA

*1 1. 10 HRS (COURSE LEVEL 100 OR 200) FROM DEPT OF MATH EXCEPT COURSE MATH 203.

3. SOCIAL SCIENCES (20 HRS)

A. COURSES TAKEN IN THIS AREA

HIST 112 S 5.0 WI79
HIST 113 A 5.0 SP73
PHIL 201 5.0 WI79

B. REMAINING REQUIREMENTS IN AREA

4. COURSES APPROP. - MAJOR (30 HRS)

A. COURSES TAKEN IN THIS AREA

CHEM 111 A 5.0 SP73

B. REMAINING REQUIREMENTS IN AREA

*1 1. CHEM 111, CHEM 112, CHEM 113 AND LABS: CHEM 116, CHEM 117 AND CHEM 118.
2. CHEM 240 AND CHEM 216
*2 4. BIO 420

5. MAJOR

A. COURSES TAKEN IN THIS AREA

B. REMAINING REQUIREMENTS IN AREA

1. CHEM 241, CHEM 217, CHEM 242, AND CHEM 218 (CHEM 218 NOT REQUIRED IN BIO OR CLINICAL CHEM OPTION.)

2. CHEM 357, CHEM 316, CHEM 358, CHEM 317, (NOT REQUIRED IN BIO OR CLINICAL OPTION) CHEM 359, CHEM 318, CHEM 480

3. REMAINING CHEMISTRY COURSES SHOULD BE SELECTED AFTER CONSULTATION WITH A CHEMISTRY ADVISOR.

6. ARTS AND LETTERS (10 HRS)

A. COURSES TAKEN IN THIS AREA

B. REMAINING REQUIREMENTS IN AREA

1. ANY OF COURSE:

ART 178 OR ART 179 OR MUS 390 OR MUS 393 OR DRAM 304

2. ANY OF COURSE:

HIST 432 OR HIST 433 OR PHIL 301 OR PHIL 302 OR ART 178 OR ART 179 OR DRAMA 304 OR DRAM 370 OR DRAM 410 OR MUS 390 OR MUS 393 OR MUS 193 OR 25 HRS (COURSE LEVEL 200) FROM DEPT OF ENG.

7. MINOR AND ELECTIVES

A. COURSES TAKEN IN THIS AREA

BIO 199 A 5.0 TRN

B. REMAINING REQUIREMENTS IN AREA

1. PHYS 237, PHYS 238, PHYS 239.

3. SELECT 10 HRS OF FOR. LANG. (102, 201) FROM FR, GER OR SELECT 10 HRS FROM IS 422, IS 260, IS 303, IS 302.

COURSES TAKEN (INCLUDING TRANSFERRED WORK) NOT APPLICABLE TO DEGREE PROGRAM.

-COURSE- GR HRS QTR -COURSE- GR HRS QTR -COURSE- GR HRS QTR
BIO 111 W 5.0 FA76 MOT 350 W 5.0 FA76 SOC 041 A 5.0 TRN
MATH 125 W 5.0 WI78

COMMENTS: *1 THIS REQUIREMENT IS PARTIALLY SATISFIED

*2 INDIVIDUALIZED COURSE REQUIREMENTS FOR YOUR ACADEMIC PROGRAM

STUDENT JOHN AVERAGE

1001 MAIN STREET

MIDWAY GA 30021

ADVISOR - JOHN P ADVISOR

GA. HIST--FAIL US HIST--PASS

GA. CONST-PASS US CONST-PASS

REGENTS EXAM--PASS

STUDENTS UNDER THE 1978-79 CATALOG SHOULD CONSULT PAGE 78 OF THE CATALOG IN REFERENCE TO THE 75 HR. UPPER DIVISION REQUIREMENT.

DELAWARE TECHNICAL & COMMUNITY COLLEGE
STUDENT RECORD SYSTEM
WHY IT IS DIFFERENT

FRANK A. KROS III
DELAWARE TECHNICAL & COMMUNITY COLLEGE

The DTCC Student Record System - It features include (1) non data processing professionals of the registrar's staff who do all their own work, (2) a sophisticated on-line inquiry, (3) over 20 computer reports, (4) table driven which has allowed for substantial change with little or no reprogramming, and (5) central updating of Tables to assure absolute uniformity among 3 Campuses of our State-wide College system.

INTRODUCTION

In January, 1976 a team was appointed to study the registration system at DTCC with the objective of revising and converting it to another computer.

The objectives of the new registration system were to:

- A. Simplify and improve registration.
- B. Reduce and simplify "drop and adds".
- C. Minimize the time spent on registration by staff, faculty and students.
- D. Provide a system run by the registrar's staff without professional data processing personnel involvement.
- E. Provide rapid analysis of information to permit deans and department chairpersons to:
 - 1. Make adjustments to class limits.
 - 2. Create or eliminate sections of courses.
 - 3. Make room and faculty changes as enrollment demands are reported.

The resultant computerized system is composed of basic elements:

- A. Master schedule preparation
- B. Registration processing
- C. Reports
- D. Grading
- E. Transcripts and grade reports
- F. Tuition calculations
- G. On-line inquiry

PHASE II

Scheduled additions to this record system are:

- A. Admissions
- B. Student Ledgers
- C. Financial Aid

The specific features discussed in this report have been in operation since Fall 1976. DTCC is currently developing the programs for Admission, Tuition and Fees Accounting (student ledgers) and Financial Aid. This paper will outline our plans for these areas.

A. MASTER SCHEDULE PREPARATION

The first step, creating a schedule of classes may be done in one of two ways: 1) rolling the course master schedule from the same quarter of the previous year, or 2) if there is no similar previous quarter, then it is better to start from scratch to create the next quarter schedule. This contains the following information:

- A. Course number
- B. Course name
- C. Hours - class, lab, and credit
- D. Fees (in-state, out-of-state)
- E. Maximum enrollment
- F. Room, time, and instructor's name

DTCC's system has many tables, one of the most important is the list of all courses offered. For each course number the table gives the course title, class, lab, and credit hours, quarter in which course was first approved, quarter in which course was inactivated (for courses no longer given), the campus for which the course is approved, the day fee and the continuing education division fee (which may or may not be the same as the day fee).

It is the responsibility of each instructional department to review the course master schedule and supply the E and F information for any course that is to be offered, while the A, B, C and D information comes from the course table.

When the departments have completed the process, the course master schedule information is sent to the registrar's office who is responsible to check all input for completeness. The registrar's office then enters the data into the computer.

The course master schedule consists of all sections of courses that the academic departments will offer for the quarter. After the master is created, a preliminary printout is sent to the Dean of Instruction. Departmental representatives then review and correct the information. The corrections are then sent to the registrar to update the master schedule. After all corrections have been processed, the final schedule is printed on xerox for mass distribution.

B. REGISTRATION

There are several levels of registration:

1. Special Registration

Certain courses offered have restricted enrollments. To enroll, the student must first obtain departmental permission. This phase of pre-registration is accomplished through advanced enrollment.

For each course that is designated as "controlled", the student contacts the department offering that course. The department lists the student's identity number on the corresponding roster. At the same time handicapped students and other exceptions are added to these lists. Upon completion, these rosters are submitted to the registrar's office for advanced registration processing.

2. Regular Pre-Registration

To successfully enroll for courses each quarter, the student must have information concerning course offerings for that particular quarter. This is accomplished through the use of the master schedule of classes. From this the student and his advisor select courses.

During the seventh week of the current quarter, a schedule of classes and registration forms (GRIDS) are distributed to currently enrolled students and mailed to all newly admitted students. The registration form is used for course requests, and payment of fees.

Once the GRID is completed, it is returned to the registrar where it is edited for any of the following possible errors:

- A. correct name and social security number
- B. correct course number

Once the registrar's office has completed the editing of the registration form, it is entered into the computer. Since fees are to be paid at this time, the student is sent to the Business Office. Payments are processed into the accounts receivable file.

3. Final Registration

For students who have not pre-registered, the final registration involves taking what's left. This group includes those who have not paid during the pre-registration period. For this group all is not lost. Another feature of DTCC's system is a reservation of any number of seats in a class by the department chairperson for students of his department. We call this a technology lock. In spite of special and pre-registration, seats may be held for those from a particular major.

C. REPORTS

1. Class lists for instructors

- a. Class lists for first day of class.
- b. Class lists are printed again following the drop-add period.
- c. Class lists are printed with grade cards.
- d. Class lists with grades are printed after grades have been processed for instructor verification.

Class lists may be printed for any specific class or classes - upon request. They may be printed for day division, continuing education division only or both. They may be printed with meeting times shown or not, with a summary of grades at the end - or not. At each printing the registrar may insert any two-line message desired.

2. Faculty report for Dean of Instruction and Department Chairperson.
3. Room utilization report for Campus Director and Deans.
4. Student lists for

Business Office
Deans
Advisors
Counselors
Veterans Office
Financial Aid Office
Admissions Office

These lists may be for a selected group such as day only, continuing education only, both, veterans only, returning students, new students, etc.

D. GRADING

Two weeks prior to the end of the quarter the registrar prints a new class list accompanied by a grade card for each student in the class.

After final exams the instructor enters the students final grade on the list and on the card. The registrar's office sorts the cards for each different grade. It is not necessary to punch the grade into each card since the cards are entered in groups and the computer program inserts the grade.

E. TRANSCRIPTS AND GRADE REPORTS

After grades are entered, the registrar runs a missing grade report. This permits a quick, simple follow-up of any missing grades. The same program can be used to "find" any grade - for example - a list of all incompletes, or failures, or A's.

When all grades have been entered the registrar runs a final class list showing the grades which are checked by the Department of Instruction.

The registrar is now ready to run the grade point average (GPA) program. This program always goes back and recalculates the current quarter and cumulative GPA since corrections and changes to previous quarter grades are possible.

At this point the registrar is ready to print the quarterly grade reports for mailing to the students.

F. TUITION CALCULATION

DTCC's tuition plan involves a specific charge for each class, a maximum tuition charge for full-time students regardless of the number of individual courses taken, and consideration of day division and continuing education division fees.

Until recently, it was not possible to correctly compute the actual tuition, due to the number of exceptions. Recent re-definitions have allowed us to program the tuition calculation and properly credit day division and continuing education division tuition earned.

Consequently, we are now planning to proceed with Phase II.

G. ON-LINE INQUIRY

Although this system is designed for further update and reporting, it also provides for on-line inquiry via terminal. This feature can be used from any terminal connected via hard wire, multiplex-leased line, or dial up telephone with acoustic coupler. A portable terminal here at this conference is set up to demonstrate the following inquiry commands:

P SSNO	obtain name, address and all demographic data on the Person whose social security number is entered.
S C QTR COURSE NO	obtains all master schedule information for the course.
S C QTR COURSE NO E	lists all the students enrolled in the course - prints grades if present in record (for past quarters).
S S SSNO D	prints a list of all quarters in which the student (whose SSNO is entered) was enrolled.
S S SSNO R QTR	lists all courses - name and course number in which student is enrolled.

These are the major inquiry commands. Access to these records is limited to approved staff members only. Obviously authorization to update the records is even more restricted than inquiry.

PHASE II (Plans)

Admissions -

Plans are to enter an applicant into the computer immediately upon receipt of the Admissions application and the \$10 admissions fee.

It is expected that the most important advantages of computerizing the Admissions process will be:

- A. Aid in reporting data on applicants acceptance and enrollments.
- B. Facilitate the use of word processing equipment in corresponding with applicants.

Tuition and Fees Accounting -

Like many new institutions we have gone through a period of less than perfect financial records.

A computerized student ledger is being developed which will accept a charge as a by-product of registration. Payments in cash, deferred or via financial aid will be credited to the student account upon input by the Business Office.

This system is in the process of development/design.

Financial Aid -

Analysis of this function has defined these areas for computerization:

- Needs Analysis
- Awarding
- Accounting for expenditures (see Tuition and Fees Accounting)
- Reporting

A GENERALIZED MANAGEMENT SYSTEM
FOR
EDUCATIONAL DATA

John S. Edwards
University of Georgia

Pier Paolo Pisani
GNOSIS, Inc.

A generalized system for managing data which is amenable to indexing for purposes of storing and retrieving was developed. This system is useful in all areas of education. The system used a data element based file structure. The file content and indexes are determined by the user at the time of creation.

INTRODUCTION

This report describes a computer-based management system for educational data which can support a variety of applications. The system is independent of the data being managed. It provides a suitable vehicle for handling data which are amenable to indexing as a method for organizing, storing, searching, and retrieving. A data element oriented file structure allows users to create any data elements needed to accurately and precisely store and retrieve data. The computer programs are table driven permitting users to specify editing, indexing and printing requirements according to the purpose of the file and the nature of the data.

File Definition. A data element file consists of records which are made up of data elements. A data element may be explained as the smallest logical and useful unit of data. Each data element may consist of individual fields separated by a field terminator. Data elements are organized into records which correspond to some logical structure of the data collection. In a bibliography for example, a record would contain all the data elements pertaining to a journal article. The data elements of this record would include author, title, keyword descriptors, date of publication, abstract, etc.

FEATURES

The Generalized Data Management System (GDMS) consists of three routines: 1) Edit, 2) Index, and 3) Master List. The following describes methods for using these routines and provides a general discussion of the usefulness and results which may be obtained from each.

Edit Routine. The Edit Routine is designed to aid in building the file which is to be indexed. The GDMS assumes that the creator of the data file is responsible for the integrity of the data. In order to assist the user, a number of edit functions may be requested from the Edit Routine. Since the creator is responsible for establishing the DEM for each file, the Edit Routine will report any record in which expected DEM are not present. The user may specify those DEM which must be found in each record and a report of any record which is deficient will be printed. The user may request a character-set edit for alphabetic alphabetic characters or numeric characters. One of the most important features of the Edit Routine checks for variant word spellings. On the user request, the GDMS will print a report of all words with five or fewer occurrences. This report allows the user to determine whether or not a word is misspelled or is used only rarely. The Edit Routine also permits automatic checking of character-set format. A DEM may be examined for exact content. For example, a DEM may contain data such as JUL1979. The Edit Routine has the ability to examine each character, determine the type whether alphabetic, numeric, or a special type, a report any variances. In the case of the DEM cited above, the Edit Routine will determine that the first three characters should be alphabetic, the next for numeric, and the final one a special character such as \$. The Edit Routine will also produce a "map" of the file in which identification numbers of each record are examined and any duplicates or skipped numbers are reported.

Index Routine. The Index Routine generates indexes from the contents of the data elements. The user specifies the quantity and type of indexes to be generated. The contents of any data

element may be used to form an index. There are five basic types of indexes which may be produced: 1) author, 2) free with a print of the complete data element, 3) free within a context, 4) keyword within context, 5) word by word cross-reference of multiple-word terms.

The author type index routine generates indexes specifically from data elements which contain data separated by terminators and in which punctuation is important and must remain in the display. In the example of figure 1, the author name is displayed with the title of the document. The number to the left of the title is the document identification which is used to locate the document in the master list. The two types of free indexes available are similar in format. The term "free index" refers to an index of uncontrolled vocabulary such as titles and abstracts. In these types of indexes, it is necessary to use a "stoplist," that is, a list of terms which are not useable as index terms. Most commonly, these are articles, conjunctions, prepositions, etc. The creator may specify terms to be included in the stoplist at the time indexes are built. In addition, the GMDS removes all punctuation from the data element before indexes are generated. An example of the type two free index is presented in figure 2. In this index the entire data element from which the index term was taken is printed. The type three free index (see figure 3) prints only a portion of the data element from which the index term was taken. The type four index is intended for controlled vocabulary data elements. The controlled vocabulary is usually taken from thesauri, authority lists, subject headings, etc. Each controlled vocabulary term is bounded by a field terminator. As shown in figure 4, the display includes all terms in the data element from which the index term was taken.

In order to demonstrate the use of an index term and all its relationships with other index terms, a type five display can be generated. As seen in figure 5, each associated term is printed along with the respective document identification number. In the case of the example, the term "AGE" is associated "AUDITORY PERCEPTION" in two documents (numbers 13 and 14), with "AURAL DISCRIMINATION" in one document, (number 22). The listing of the term "AGE" four times indicates that this term occurs in four documents in this collection.

The index routine generates cross-references from among any data elements or fields of data elements or defined by the user. This ability to generate an almost unlimited number of cross-references at the request of the user provides the power and flexibility to accommodate

diverse types of content.

Master List. The master list is a complete listing of the data file. The user may specify the data elements and their order at the time of printing.

USAGE

The most extensive use of the GDMS has been to manage a large collection of competency statements for the College of Veterinary Medicine at the University of Georgia. Approximately five thousand statements were defined by statement characteristics such as required behavior and criteria of acceptance. In addition, each statement was described with terms reflecting discipline, organ system, animal, academic course and other terms pertinent to Veterinary Medicine. The indexes generated have provided a useful means for faculty to easily determine the objectives of the courses they teach. The students benefit by having ready access to behavior expected in individual courses and for programs of study and the entire curriculum leading to the academic degree.

Many faculty are using the GDMS to generate indexes of private and departmental libraries. These projects have been carried out in such diverse areas as Music, Plant Pathology, Biology, and Animal Science.

One of the most intriguing uses is the analysis of an American Indian language. In this project, the Indian language sentences were entered along with their translated meanings, morphological categories, and speaker location and age. The indexes facilitated the analysis of the language structure and the building of an annotated dictionary.

BAIRD, SALLY
0000023

A technique to assess the preference for intensity of musical stimuli in young hard-of-hearing children.

DE YARMAN, ROBERT, M.
0000001

An experimental analysis of the development of rhythmic and tonal capabilities of kindergarten and first grade children.

FIGURE 1. AUTHOR INDEX

RHYTHMIC
000000001

An experimental analysis of the development of rhythmic and tonal perception of kindergarten and first grade children.

FIGURE 2. FREE INDEX TYPE 2

RHYTHMIC
000000001

Development of rhythmic and tonal perception.

FIGURE 3. FREE INDEX TYPE 3

AESTHETIC SENSITIVITY

0000008

Second grade; third grade; experimental research; aural discrimination; timbre; rhythm; form; aesthetic sensitivity; cognitive ability; aesthetic experience; television.

FIGURE 4. CONTROLLED INDEX TYPE 4

AGE

AGE	.0000013
AGE	.0000014
AGE	.0000022
AGE	.0000027
AUDITORY PERCEPTION	.0000013
AUDITORY PERCEPTION	.0000014
AURAL DISCRIMINATION	.0000022
DESCRIPTIVE RESEARCH	.0000013
DESCRIPTIVE RESEARCH	.0000013
DESCRIPTIVE RESEARCH	.0000014
DESCRIPTIVE RESEARCH	.0000027

FIGURE 5. TERM ASSOCIATION INDEX TYPE 5

PROJECTING ENROLLMENTS FOR ELEMENTARY AND SECONDARY SCHOOLS

William J. Hierstein
The University of Iowa

The advent of the phenomenon of declining enrollments in virtually all elementary and secondary school systems has led boards of education and administrators to call for accurate, or at least credible enrollment projections to assist them in making important management decisions. This has raised some questions regarding the adequacy of traditional forecasting strategies and has pointed to a clear need for further investigation of the subject. The IENROL program provides the user with a number of options for enrollment projection. The findings of a preliminary investigation show that adequate enrollment projection strategies can be developed by careful combinations of established statistical techniques and data processing systems.

School executives, perhaps today more than ever before, are quite concerned with population trends. This concern is motivated by more than the perfunctory management interest in collecting data on which to base decisions and make future plans. The fact is that during the last decade school systems have found themselves caught in the middle of a population trend that was largely unanticipated by educational planners of the late 1950's and early 1960's.

In 1950 there were 28 million pupils in the elementary and secondary schools. By 1960 there were 42.7 million school age children and the prospects were bright for even more pupils in the 1970's (Sargent, 1975). This was thought to be true especially because the post-war baby boom children had come of age and were beginning to have families of their own. Few people anticipated that the dip in the birth rate that began in 1957 would be anything but temporary. The birthrate, however, continued to decline steadily until 1968 and remains today substantially below the 1957 level.

The reduced number of live births could not help but affect school systems adversely. The late '70's have witnessed grim realities of fewer children in the schools. School district boards and their constituencies in many systems know the turmoil that accompanies declining enrollments. Communities have slowly learned that as the number of pupils declines so does the need for teachers, administrators, and classroom space. District after district has been forced

to close buildings or classrooms, release teachers and administrators, and face the loss of funds that are a function of enrollment.

School managers today need to have accurate or at least credible forecasts of future enrollment to aid them in making the difficult decisions associated with declining enrollments. Questions requiring answers supported by intelligent and reliable forecasts include: (1) should buildings be closed? (2) once closed should the building be sold, "mothballed", or leased? (3) must innovative programs be cut back or can they be maintained without damage to the central program? (4) will drastic staff reductions be necessary next year? (5) is there any end in sight to the enrollment decline?

Armstrong and Groham (1972, pp. 211-213) identify two dimensions of forecasts: (1) the subjective vs. objective dimension, and (2) the naive vs. causal dimension. Subjective forecasts are those in which the process used to obtain the forecast has not been well specified and in which the process is usually carried out "in the head" of the forecaster. Objective forecasting methods, on the other hand, are those in which the process used to obtain the forecast has been well defined (hopefully so well defined that it can be replicated by others). Naive methods are those which use only data on the dependent variable and in which the analysis attempts to identify regularity in the dependent variable over time. Causal forecasting goes beyond the dependent variable to

260 consider also variables which may cause changes in the dependent variable.

Many administrative decisions are based on subjective forecasts. Decisions which rely on the personal experience and intuition of the decision maker often prove to be sound. There are in fact some circumstances in which subjective forecasts are clearly superior to objective forecasts. An example of such a circumstance, as it affects enrollments, might occur when a large new industry moves into a community or an old one moves out. There are, however, other instances in which subjective forecasts have led decision makers into disaster. This point is illustrated by the fact that some school districts continued to construct new buildings to house increased enrollments that never occurred. Generally it can be said that though decisions sometimes have to be made based on subjective forecasts, objective forecasts are preferable.

Objective forecasts lend themselves to EDP techniques and are therefore of most interest to educational data processors. The fact that objective forecasting techniques are based on statistical procedures or algorithms and the data sets are often extensive and require careful analysis, make the computer a necessary part of objective enrollment forecasts.

Most of the objective techniques commonly used to project elementary and secondary enrollments can be described as objective-naive methods, in that they attempt to identify trends in enrollment over time and do not attempt to consider the effects of other variables that may have a causal relationship to enrollment changes. Probably the most common method is the cohort survival ratio technique, which calculates both initial grade data (enrollment forecasts for the lowest grade in the data set) and survival ratios (forecasts utilizing initial grade data that project enrollments based on the ratio of change from prior year and prior grade) by determining the ratio of enrollment fluctuations over time. More recently regression techniques have been applied to produce enrollment projection. Although the techniques differ greatly, they can still be described as objective-naive, because they consider only the relationship of enrollment (the dependent variable) to time.

Causal forecasting techniques differ from naive techniques in that they consider variables that have a "causal" relationship to enrollment when calculating the initial grade data. Statistics such as the number of live births in preceding years, housing starts, and in-

out migration estimates have all been used by researchers attempting to obtain more accurate enrollment projections. There is at present little evidence that objective-causal methods are superior to objective-naive methods.

The key to accurate enrollment forecasting seems to be careful selection of a forecasting technique. Apart from the differences that can result from the decision to either include or exclude from consideration causal factors, there are also differences that result from the application of various statistical techniques.

IENROL is a computer program that was developed to provide the user with a wide variety of options for enrollment projection while maintaining simplicity of operation.¹ The program is written in transportable BASIC and is designed for interactive use.² It can be adapted for execution on microcomputers without substantial modification. Furthermore, only a fundamental level of computer literacy is required of the user.

The large number of options available to the user and the opportunity to use both objective-naive and objective-causal methods make IENROL a powerful tool for decision making. Ease of operation and relatively simple programming make it suitable for use by a wide variety of users including those who are unfamiliar with computer operations. Research and evaluation of the various options available in IENROL are continuing. At this point, it is not appropriate to speculate about the superiority of certain methods over others. However, early results do show significant differences in the forecasts obtained using the various combinations of options in IENROL.

After submitting a background data set, the IENROL user may select to have initial grade data calculated by any one of the following options:

- (1) MEAN ARITHMETIC CHANGE
- (2) MEAN PERCENTAGE CHANGE
- (3) MEAN ARITHMETIC CHANGE WITH DEVIANT YEARS EXCLUDED
- (4) MEAN PERCENTAGE CHANGE WITH DEVIANT YEARS EXCLUDED
- (5) REGRESSION - ENROLLMENT AS A FUNCTION OF TIME
- (6) CENSUS DATA TRANSFORMATION
- (7) USER SUPPLIED INITIAL GRADE DATA

The program will calculate initial grade data from a minimum of three years of background data and will calculate initial grade data for the number of years that the user wishes to forecast.

Enrollment projections may be calculated using IENROL by combining any one of the methods currently available with the initial grade data method selected. The methods available for selection are:

- (1) MEAN SURVIVAL RATIOS
- (2) MEAN SURVIVAL RATIOS + A PORTION OF THE STANDARD DEVIATION
- (3) MEAN SURVIVAL RATIOS - A PORTION OF THE STANDARD DEVIATION
- (4) SURVIVAL RATIOS USING A LOGISTIC CURVE
- (5) REGRESSION - SURVIVAL RATIO AS A FUNCTION OF TIME
- (6) INTRODUCED SURVIVAL RATIOS
- (7) REGRESSION - ENROLLMENT AS A FUNCTION OF TIME
- (8) REGRESSION - ENROLLMENT AS A FUNCTION OF PRIOR GRADE ENROLLMENT
- (9) REGRESSION - ENROLLMENT AS A FUNCTION OF PRIOR GRADE AND PRIOR YEAR

Although most of the options available are probably familiar to the reader, a few words of explanation concerning the logistic curve technique may be appropriate. Projections determined by use of a logistic curve are generated by the function

$$P(x) = k_1 + \frac{k_2 - k_1}{1 + e^{a+bx}}$$

where k_1 and k_2 represent acceptable limits and a and b represent coefficients of regression using a natural logarithm transformation.

The large number of options available to the user and the opportunity to use both objective-naive and objective-causal methods make IENROL a powerful tool for decision making. Ease of operation and relatively simple programming make it suitable for use by a wide variety of users including those who are unfamiliar with computer operations. Research and evaluation of the various options available in IENROL are continuing. At this point, it is not appropriate to speculate about the superiority of certain methods over others. However, early results do show significant differences in the forecasts obtained using the various combinations of options in IENROL.

Notes

1. IENROL was developed as a seminar project in Computer Applications conducted by Dr. William C. Bozeman at The University of Iowa in the fall of 1978.
2. The term transportable BASIC refers to the fact that the programmers attempted to use programming

procedures that would facilitate utilization of the program on a wide variety of computer hardware.

3. The derivations and a discussion of this method may be found in many statistics textbooks under the topic of regression coefficients in a two variable linear equation.

Footnotes

Armstrong, J. Scott, & Grohman, Michail C. A comparative study of methods for long-range market forecasting. Management Science, October 1972, 211-221.

Sargent Cyril G. Fewer pupils, surplus space: the problem of school shrinkage. Phi Delta Kappan, January 1975, 352-357.

SCORE: A TEST SCORING AND ANALYSIS PROGRAM

Edwin J. Jauffmann, Jr.
Corporate Time-Sharing Services, Incorporated
(Formerly, St. John's University, New York)

Robert L. Williams
St. John's University, New York

SCORE is a data analysis program that analyzes test scores placed in a terminal format file from mark-sensed answer sheets processed by a 3M 5500 Data Analysis Machine (DAM), a program executed at the DAM and a hard-copy terminal by individual instructors without technical assistance. Composed of seven "chained" programs, SCORE statistically treats raw scores to produce percentile ranks, standard normal Z scores, "curved" scores, and T scores. For each test item, a raw item analysis is produced along with "difficulty" and "discrimination" indices. Handling up to 300 items for up to 300 students, SCORE, written in DTSS BASIC, operates efficiently.

Most academic computer applications at St. John's University in the past decade have been conducted via time-sharing services. At this time faculty and students have access to the services of two time-sharing companies through 40 terminals located at various academic stations; the principal one being the Academic Computing Laboratory--essentially a center with 20 terminals dedicated to student computing--; the others being one- or two-terminal stations in a number of academic departments principally dedicated for faculty use. In the College of Pharmacy and Allied Health Professions several applications are receiving attention: computer assisted instruction; drug information systems; test construction programs; and test scoring programs have been developed and are being extended.

Test construction and test scoring programs, like SCORE, are being developed in the Pharmacy Education Resource Center (PERC), a "library-like" unit which currently makes available to students and faculty five CRT and three hardcopy terminals. The Director of PERC assists faculty; PERC is not staffed in a manner to "do" or "take over" faculty projects. It is expected that faculty members must contribute significantly to their own projects. However, PERC has sought to help relieve faculty of some of their more repetitive instructional tasks. Test construction--actually test production--and test scoring are two tasks that are now receiving attention. Yet, in trying to give assistance to the faculty, the attempt has been made to provide the

faculty tools which they can use independently without technical assistance on the hardcopy terminals in PERC. Such is the case with SCORE.

SCORE is a data analysis program that analyzes test scores placed in a terminal format file from mark sensed answer sheets processed by a 3M 5500 Data Analysis Machine (DAM) located in PERC, a program arranged to be executed by individual instructors without technical assistance.

At the time an examination is given, the instructor requires his students to use mark sensed answer sheets. (SCORE assumes multiple-choice, yes-no, or true-false questions.) Each answer sheet bears a student identification number. Two methods of identification are available, social security number or roster (seat) position. Both create grade listings that may be posted, the latter (roster or seat position) facilitating an easier transfer of grades to roll books.

The DAM is connected to a hardcopy terminal. First, the instructor feeds his answer key, or keys, through DAM. Second, he feeds his students' answer sheets through DAM. Then, as the sheets are accepted DAM marks incorrect answers and sums the number marked correctly, printing that sum at the side of the answer sheet. The students' responses and I. D. number are simultaneously stored in a terminal format file. At this time the answer sheets of a class

have been corrected and bear raw scores.

The instructor now 'executes' SCORE, actually a series of seven 'chained' programs that operate on the terminal format file created by DAM, along with other files. To increase efficiency and to reduce core requirements, SCORE was programmed to take advantage of the 'system sorter' and program 'chaining' of the Dartmouth Time-Sharing System (DTSS) available at St. John's through Corporate Time-Sharing Services. Chaining, or segmentation, allows a division of labor, as it were. This is not apparent to the user, however. The user responds to a series of questions in one sitting, giving SCORE information to be used in its various programs.

When the instructor 'executes' SCORE, the first segment passes the current file created by DAM to the system sorter, which sorts on the first ten characters of each record--student I. D. and page number, eliminating all duplicates, if any. At this time, the second segment operates, and the instructor receives at the terminal two questions to respond to:

How many answer sheet columns for an individual test? 1

The total number of questions for the test is? 35

With the instructor's responses, this second segment then assembles the multiple pages (if more than one) of the students' answers into one record for each student, the record being composed of the student's 9 digit I. D. number, and his responses to all questions. In this second segment the answer key data is stored in a separate 'scratch' file. Also, in this second segment, should any answer sheets fail to be processed into one record, they are eliminated from the analysis and flagged to the instructor via the terminal.

The third segment now 'grades' each valid test and assigns a raw score to each I. D. number. Summary statistics that include (1) number of valid tests; (2) mean; (3) standard deviation; (4) standard error of the mean are also compiled at this time through rather straight forward procedures.

The file contains the raw scores in ascending numerical order to the fourth segment. At this time, using the upper and lower 27% of the total raw scores--the middle 46% of the scores being ignored--frequency counts for each response to each question are summed.

Raw score frequencies for all students are compiled. This frequency table of raw scores, which is actually a random access numeric file, is used to calculate percentile ranks for each raw score, the first step toward 'curved scores.'

The fifth segment then finds standard normal Z scores by 'interpreting' percentile ranks through a table search. An item analysis using previously compiled frequency statistics is also generated, consisting of two parts--(1) an item difficulty index and (2) an item discrimination index.

An item difficulty index is one that ranges from .00 to 1.00.

Number who answered correctly

Number who tried the item X 100

$$Dif = R/T \times 100$$

An item discrimination index is one that may be either positive or negative, usually expressed as a decimal. If zero, there is no discrimination. If negative, the indication is that the less knowledgeable students tended to answer correctly. If positive, the indication is that the more knowledgeable students tended to answer correctly. 'Upper' (U) indicates students who have high, overall scores--the top 27%; 'Lower' (L) indicates students who have low scores--the bottom 27%.

Upper students - Lower students

One-half of Upper + Lower Students

$$Dis = (U-L) / (1/2) (U+L)$$

(Mehrens, William A., and Irvin J. Lehmann. Measurement and Evaluation in Education & Psychology, Holt, Rinehart and Winston, Inc., New York, 1973.) Usually, the higher the discrimination index the better. But there can be instances in which this may not be desired for an item or set of items.

Even without "using" the indices the instructor can often make reasoned judgments about his examination by visually screening the raw frequency counts of responses. Following a visual

screening; attention can be given to the indices of selected items that have 'caught the eye.'

The sixth segment asks the instructor to choose one of two options:

There are two options for curved scores.
Option one (1) allows you to specify a mean and upper bound. From this information the program will calculate a standard deviation so that, when curved, the scores will fit your specifications.
Option two (2) allows you to specify a mean and standard deviation which will be used directly to curve your scores.
Please indicate your preference by typing 1 or 2 in answer to the following question.

Curve option

7 1

Type the mean and upper bound you wish the curved scores to have? 75:100

The first accepts from him an arbitrary mean and an arbitrary top score the class is to have assigned to it; the top score being one a student would receive if theoretically at the 100th

percentile. The second accepts from the instructor an arbitrary mean and standard deviation. For both options the raw scores are statistically 'normalized.' From the option selected in this sixth segment and the curve parameters input by the instructor, 'curved' scores are generated from standard normal Z's obtained earlier in the fifth segment. A summary of terms describing the printout to follow is presented to the user at this time.

The seventh segment now prints out the computed array of data for the instructor after he has set the printhead to the top of the 'next page.' The array consists of a roster, an item analysis, summary statistics, and histograms as shown in the following portions selected from an example:

Test Analysis				09/01/78	09:28:18	Page 1	
Student #	Raw	% Score	Rank	z Score	Curved	T Score	
000000001	Incomplete		Incomplete				
000000002	17	48.57	13.97	-1.0816	66.0	392	
000000003	29	82.86	98.53	2.1780	93.2	718	
000000004	12	34.29	2.21	-2.0121	58.2	299	
000000005	18	51.43	17.65	-0.9287	67.3	407	
000000006	14	40.00	5.88	-1.5649	62.0	344	
000000007	29	82.86	98.53	2.1780	93.2	718	
000000008	8	22.86	0.74	-2.4372	54.7	256	
000000009	19	54.29	22.79	-0.7457	68.8	425	

Test Analysis		09/01/78							09:28:18		Page	4
Question Number		0	1	2	3	4	5	>5	C Ans & Dif			
									Dis-Ind	Ind		
U	1	0	0	0	1	1	16	0	5	63.89		
L		0	7	1	1	2	7	0	0.50			
U	2	0	0	0	0	0	18	0	5	91.67		
L		0	0	0	1	2	15	0	0.17			
U	3	0	1	13	4	0	0	0	2	52.78		
L		0	3	6	6	3	0	0	0.39			
U	4	0	12	0	0	1	5	0	1	52.78		
L		0	7	4	0	1	6	0	0.28			
U	5	0	0	11	7	0	0	0	3	30.56		
L		0	1	13	4	0	0	0	0.17			
U	6	0	15	0	2	0	1	0	1	68.57		
L		1	9	3	4	0	1	0	0.34			

Raw Score Histogram

```
8 *
9
10
11
12 *
13 *
14 **
15 *
16 **
17 ***
18 **
19 *****
20 *****
21 *****
22 *****
23 *****
24 *****
25 ****
26 ****
27 ***
28 ****
29 **
```

STATISTICAL ANALYSIS

NUMBER OF VALID TESTS 68

ARITHMETIC MEAN 21.7353

STANDARD DEVIATION 4.16787

STANDARD ERROR 0.505428

RANGE 8 - 29

For a set of 48 examinations involving three answer columns with a total of 100 five-part multiple-choice questions, an average session with SCORE requires approximately three-quarters of an hour. Generally, about 30 minutes are needed to input the answer key and student answer sheets; it being expected that a number of answers would be

incorrectly marked by a student or two, rejected, and 'corrected' by the instructor and re-entered. (A careful review of answer sheets by the instructor before entering them is suggested.) The actual statistical calculations by SCORES' segments require approximately 1 minute of real time. The printout, in turn, requires about 12 minutes.

A SYSTEMATIC INFORMATION MANAGEMENT MODEL FOR COOPERATIVE EDUCATION: DATA BASE DEVELOPMENT & EVALUATION

Robin D. Avery, Ph.D.
Jeanne Dulaney-Sorochak

Macomb County Community College

In a one hour presentation, the intent is to describe the current status of MCCC's Co-op Management Information System (OMIS) as it exemplifies the local effort and the national model for similar programs. To highlight features of the system, there will be an on-line demonstration and presentation of various system components. The main focus will be on how educators can develop and refine similar information systems using their local data resources.

I. Introduction

Management Information Systems (MIS) and Data Base Management Systems (DBMS) have become increasingly important in instructional programs and administration of modern education. Techniques for applying these computer-based tools to educational installations (as opposed to their business counterparts) have been discussed widely in recent years. Today many systems are being developed, refined and exported on an international scale.

Major applications of computerized information management, however, tend to remain in the business offices and registration areas of educational institutions. (And while many instructional applications have been produced and carefully refined, instructional computing has been omitted from this discussion.) Nonetheless, many units of an educational institution can benefit from the development of specialized Management Information Systems and data bases. These discrete MIS and DBMS applications need not be incompatible with or even separate from a larger, more universal system in use in the institution as a whole. However, often the information which is essential to the operation of one unit needs to be developed separately from the information which services the rest of the institution. It is under this premise that the Co-op Management Information System (OMIS) is being developed at Macomb County Community College.

Macomb's Cooperative Education Program has its specialized information needs, as do all other instructional, service, and administrative areas of the college. But one of the major advantages of the OMIS is that it collects and stores information for staff utilization, much of which is of no interest to the rest of the college. At the same time, the system provides for utilization of information available for general use from the larger data base, thereby minimizing data redundancy wherever possible. Obviously much work remains to be done before a comprehensive, integrated system can be demonstrated which is consistent

with the larger college data base structure. But one of the major goals of the OMIS is to design such a system, specifically for cooperative education, which has the capability to interface with other areas -- both internally (at MCCC) and externally (at other schools). Thus the OMIS is being designed as a national (possibly international) model for use in various environments at schools and colleges with existing cooperative education programs.

In order to clarify the specialized needs of the Co-op environment, a brief working definition of the general concept is useful. Cooperative education is a process by which the student attends classes which lead to the completion of a pre-designed program. While attending classes the student may be employed on a part time basis in a field related to his or her academic program (Parallel); or the student may rotate between periods of full time employment and full time study (Alternating). What makes the Co-op experience different from normal placement or "work-study" is the coordination between the school and the employer, resulting in academic credit earned (and usually counting as elective and/or major credit towards graduation). Within this definition many variations are possible. The student may be involved in various programs: liberal arts, engineering, pre-professional, business, public service, education, etc. Furthermore, in recent years the Co-op experience has expanded to include a wide range of educational levels: community colleges, post-secondary technical schools, colleges, universities, and graduate schools. A wide variety of certificates and degrees are now possible involving both voluntary and mandatory cooperative education programs.

From this definition, then, it became clear that the Co-op unit had information needs which differed from those of the total school because of the primary types of information generators/consumers: students, employers, and educators. In addition, cooperative education has experienced rapid growth within the last fifteen years, requiring additional coordination of academic/employment

information in order to provide timely delivery of appropriate information to appropriate individuals or groups. The result of this combination of specialized information needs and rapid program growth has generally been a chronic information management problem at most schools. In 1978 MCCC received a federal grant to develop CMIS as a national model. The major national intent is to develop and field test CMIS at MCCC and other selected schools nationwide. If successful in the various test environments, eventually (by the mid 1980's) many schools would be able to adopt the CMIS framework in its refined form(s).

A basic assumption of CMIS is the necessity to develop improved methodologies to determine, collect, store, manipulate, and retrieve information pertaining to the Co-op experience. In order to structure this design criterion realistically, the following project goals and guidelines were developed: to

1. develop clear and simple procedures so that the system can be maintained by non-computer-oriented personnel.
2. provide for accessibility of information for planned and unplanned needs without relying on sophisticated computer programming support.
3. supply timely information to satisfy diverse users.
4. develop procedures and techniques which are non-machine dependent so that CMIS can be easily exported to environments with different computer hardware than MCCC or to environments with no computer hardware.

II. The Co-op Management Information System Model and Data Base Development

The CMIS model consists of an overall framework, a visual representation of the operation of the Management Information System (see Figure 1), additional methodologies and procedures for the installation of CMIS at individual institutions, examples of how each component has been implemented at MCCC, and descriptions of how these components may be modified for use in different Co-op environments. There are four major components of the CMIS: A. Information Need Analysis, B. Data Collection, C. Data Manipulation and Storage, and D. Information Reporting. All of these components are essential to the development of an effective computerized and/or non-computerized CMIS.

- A. Information Need Analysis: This step is a primary determinant of the original content of the data base. That is, in order to structure a data base comprehensively it must be determined what people perceive as the essential information needed to perform their jobs efficiently. Thusfar no one method has been able to determine all perceived information needs. Employer and educational practitioners have been inconsistent and vague in reporting the types of information used. Furthermore, as various needs were reported, additional needs were expressed; and as program needs change, it appears that informa-

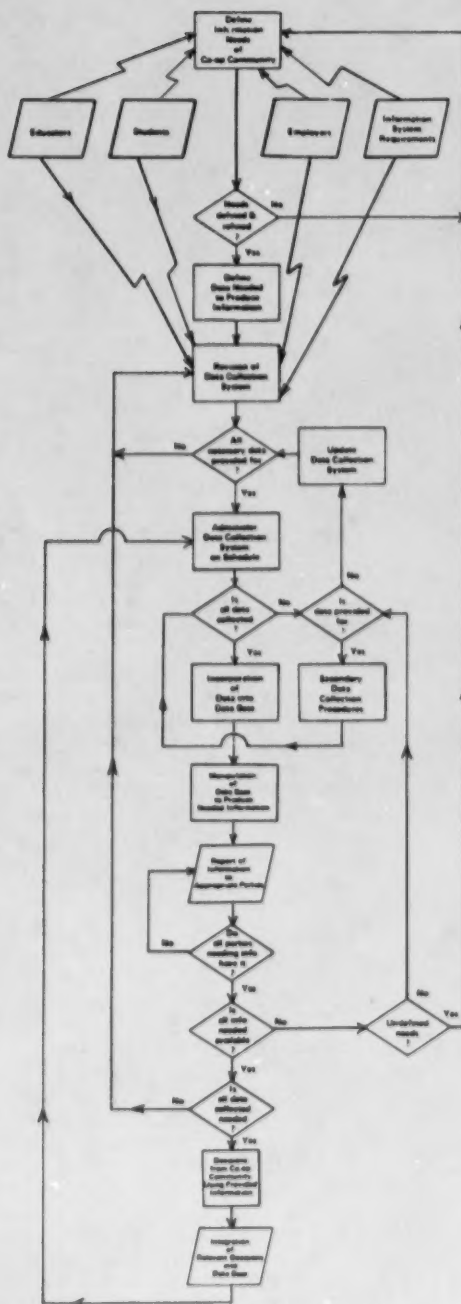
tion needs also change. Therefore, it would seem that Information Need Analysis must be an ongoing process (dynamic rather than static) and can only be assessed through a combination of several methods.

At Macomb four methods were used for initial information need analysis: an information need survey, discussion with Co-op practitioners, monitoring frequently mentioned topics in staff meetings, and discussions with national Co-op authorities. Practitioners were asked to respond to various information lists produced by the project staff. These lists stimulated discussion in and out of staff meetings, with Co-op employers and throughout the local MCCC environment. Discussions with national Co-op authorities occurred during staff attendance at various regional and national conferences, and several individuals were hired as consultants to aid in the structuring of appropriate information need analysis instruments. After the initial list of information needs was determined and summarized, it was analyzed and formatted for efficient production.

- B. Data Collection: Data are collected in any system, whether manual or computer-based. CMIS concentrates on developing consistent procedures for getting information into the system efficiently. In this effort two goals have been emphasized simultaneously: first, to maximize efficiency and to avoid redundancy in determining the value of existing collection instruments; second, to analyze where, when, and from whom information is most efficiently collected. The initial result has been a package of forms which collect necessary information for Co-op staff and employers about all elements of the Co-op experience. Further, reevaluation of this package assists in the refinement of the data base. That is, once a collection cycle has been completed, a review of collected data allows for the revision of information needs. Further, collection procedures may limit what data are practical to collect and in what form(s). This continuous review process, then, reduces redundancy of information collected.
- C. Data Storage and Manipulation: In order to maximize access to and manipulation of data, the following data base has been designed and is under development at MCCC: Current Student Records, Former Student Records, Employer Records, and Unit Operations Records. In many cases the various records include formatting of data fields which are coded and linked in a similar manner to that of the main college data base (IDS) to insure compatibility. In other cases, when it has been determined that no later connection will be necessary, information and data files are constructed and modified without regard to predefined structures. This allows for flexibility and experimentation with the CMIS and data base design; thus evolvability of the system is enhanced.

However, certain constraints do exist and have a significant impact on CMIS development at MCCC. Internal data processing problems,

Figure 1 Conceptual Schema of a Co-op Management Information System (CMIS)



1 Before an information system can be established, needs for information must be determined. This is done in several ways: information need surveys, analysis of current data collection and information reporting systems, and informal discussion with those involved.

2 Major Strands of those involved in Co-op. These groups should be involved in determining their own information needs, how the data needed is to be collected, and the form it is to be reported in. Where decisions are made, the interests of these groups must be considered.

3 Part of the development of the information system is ending in the process of defining needs. Many of those involved in co-op need aid in determining exactly what information is needed.

4 Information needs must be broken into the kinds of data needed to produce it. (See glossary for definition.) Data needed must be analyzed for availability and collectability.

5 Using data analysis and input from those using the data collection system, existing data collection system should be revised to collect data in the most efficient method possible. (See appendix F for current steps of revised forms.)

6 Continuous review and update of the data collection system ensures that it remains complete and responsive to the needs of information users.

7 Data collection systems for Co-op have a time frame which is set by policy. This includes deadlines for applications, time frames for student goals and evaluations, etc. A time frame should have as much flexibility as possible while still allowing information to be completed when needed.

8 Data should be as complete and accurate as possible and all data needed to produce information should be available. Students, employees, and educators should be encouraged to be as complete as possible.

9 Data base entry should be easily accomplished, preferably without using key punch, to update the data base quickly.

Data collection systems should have a back up system to collect data which is not collected during regular administration.

10 Data base should be useful in both batch reporting and on line manipulation to give maximum flexibility to the CMIS.

11 Reporting, including batch and on line manipulation of the form in which information is reported to information users, should be developed in consultation with those users to assure information is in most usable form.

12 All possible information users should have easy access to necessary information. System personnel should be aware of possible overights and provide reports to those users upon requests. Requests should be noted as continual or one-time needs, and if continual, should be incorporated in future reporting schedules.

13 Information needs must be continuously reviewed to determine if the information needs of the co-op community are still being met by the system. Additionally, data which is no longer needed to produce information and shows no necessity for future reporting or is collected by some other method should be dropped from use, as a system which is continually growing becomes unwieldy.

14 Information provided by CMIS is used to aid decision making by co-op participants. This is an auxiliary function, utilizing information from the system but not controlled by it.

15 Many of the decisions about co-op have implications for the CMIS: types of information needed in the future, ways of reporting, policy and procedural changes, etc. These should be incorporated where necessary and the system should be continued by administering data collection instruments on schedule.

most important of which are lack of sufficient systems support personnel and incomplete conversion programming relating to an earlier major configuration upgrade (IBM to Honeywell), forced the decision to maintain a separation between CMIS and the main data base (an IDS system). Also storage limitations have required temporary use of external data processing for manipulation of some Former Student Records. Thus the extent of manipulation of data -- internal and external -- is currently determined largely by management reporting needs. Therefore, it is not possible to project accurately the future development of data integration or data independence of CMIS at this time.

D. Information Reporting: CMIS information reporting is accomplished in three basic ways:

1. Summary reports provide information regarding student characteristics, work performance evaluations, employers' needs, etc. on a periodic basis. 2. Information lists are provided manually and on computer. The manual lists provide off-line access at all times. These are particularly useful when computer access is impractical. The need for computerizing non-mathematical information (alphanumeric lists) to provide coordinators and employers with quick access to selected groups of categorical information through terminal access has been partially accomplished through various procedures of string manipulation. An efficient computerized system must have this capability as well as the option of accessing student records in a timesharing format (spontaneous query capability). The CMIS at MCCC is developing in that direction. 3. Finally, information reporting in CMIS responds to unplanned needs. Coordinators, MCCC administration, Co-op employers, and students request information on a need-to-know basis, often for the first time. These requests, traditionally processed manually, are increasingly adapted to computerized formats. However, this tendency towards computerization in response to unplanned needs is not yet well documented. Some miscellaneous needs are satisfied from existing timesharing lists, but many are still handled manually. Nonetheless, the CMIS data base design is flexible enough to allow growth in both computerized and non-computerized applications. And in some cases both may be desirable, one to provide backup for the other.

III. Conclusion

Macomb County Community College's Co-op Management Information System is in the development stage, with the aid of local cooperative education practitioners and national experts; eventually it is planned to export the CMIS model to other cooperative education programs nationally, with or without the aid of computers. If this plan is successful, the model will provide certain advantages to the various participating institutions. The techniques used to implement, document and evaluate CMIS will allow institutions to survey information needs, design appropriate data collection systems, design and

refine separate, parallel, or subordinate data bases, report information efficiently, and coordinate program level systems with larger institutional systems.

For schools that have no MIS development, CMIS can provide the framework for total MIS system design. The use of efficient management information tools aids the operation of any unit, and careful documentation of procedures and techniques can be beneficial to all levels of an organization. Thus it is hoped that Macomb's CMIS will be regarded as a useful addition to the area of Educational Data Base and Management Information Systems.

REFERENCES

- Frankel, Steven, Cohen, Alan J., Deane, Robert T. Cooperative Education-A National Assessment: Abridged Final Report. Silver Spring, MD: Applied Management Sciences, Inc., 1977.
- Honeywell Information Systems. Time Sharing Text Editor. Waltham, MASS: 1975.
- Martin, James. Computer Data-Base Organization. New Jersey: Prentice-Hall, Inc., 1977.
- Ross, Ronald G. Data Base Systems: Design, Implementation, and Management. New York: ANACOM, 1978.
- Senn, James A. "Essential Principles of Information Systems Development." MIS Quarterly, (June, 1978), 17-26
- Stern, Miles. "Data Base Concepts within the MAS Curriculum." SIGCSE Bulletin, Vol. 10, No. 1, February, 1978.

USING A COMPUTER TO AID IN EVALUATING THE OCCUPATIONAL ACTIVITIES OF A STUDENT'S PROGRESS IN A PHYSICIAN'S ASSISTANT PROGRAM¹

By

Jesse Edwards
Thomas Gallagher
Valgene Valgora

Leonard Campbell
D. F. Costello

Physician's Assistant Program
University of Nebraska
Medical Center
Omaha, Nebraska

University of Nebraska
Computer Network
Lincoln, Nebraska

The paper describes the process of interaction between University departments and the University Computer Network as they jointly attempt to develop a computer program to aid in the curriculum evaluation process. The system design aspects, the choice of equipment, the choice of an implementation language and the resulting systems will be described and evaluated.

The first three authors are administrators of the Physician's Assistant Program at the University of Nebraska College of Medicine and were responsible for system design and evaluation. The latter two authors will describe the equipment configuration considerations leading to the choice of an IBM 5110 mini-computer, the choice of APL as the programming language and the possible implications of these decisions on future developments.

SUMMARY

The Physician's Assistant Program is a relatively new concept that has been operating in the United States for less than fifteen years. The ideal course of study that gives rise to reasonable certification procedures for physician's assistants (PA's) is still evolving. In the training period the PA spends time in the field working under the tutelage of a licensed physician. In order to inventory, measure and analyze the occupational activity of these students, a system was designed to classify and grade these experiences. A computer program was then developed which records the progress of the individual student on a month to month and cumulative year-to-date basis. The system, the coding, the management reports and the block diagram description of the computer project involved are described herein. The programs were written in APL and run on an IBM 5110 computer system.

INTRODUCTION

In the development of new experiential based programs, curriculum designers rarely know what to include and exclude from the educational experience of the student.

Given the newness of the PA program, it was soon obvious that an occupational analysis of the PA program was mandatory. Many consumers in the PA concept have an intense interest in this development.

"Health planners need to know about the capabilities and functions of a PA so that the PA concept can be included in the planning process. The general public, as patients, needs to know what PA's are doing and capable of doing. Curriculum planners and program directors should find a detailed occupational analysis of the PA to be of great value in planning educational activities for PA's. Physicians who are considering the employment of a PA should find such an analysis to be of value in making hiring decisions. Prospective student PA's would use the information in making career decisions. All members of the allied health professions should be interested in this new addition to the health profession. Perhaps the most interested group is the PA profession; as individuals they should find it useful to compare their employment and activities to other PA's. (Ref. 1)

¹This paper describes a joint project between the University of Nebraska Medical Center's Physician's Assistant Program and the staff of the Academic Computing Services directorate of the University of Nebraska Computer Network. The project was funded by a grant from the University of Nebraska Academic Computing Service's Task Force as well as funds made available by the Dean, University of Nebraska College of Medicine.

The detailed discussion of the research behind the substantive investigation of the PA program is beyond the scope of this paper. It suffices to say that the information generated from the relevant data collection and analysis should help answer such questions as: What kind of patients do PA's treat? What conditions do they see? At what intellectual level are they involved in the treatment program? What conditions don't they see? Are any patterns evident?

Perhaps the most important output from the project will be the continuing and evolving record of what the PA's learn in the field and how and to what degree they are utilized. It is also important to note that since a PA returns to the classroom after the field experience, obvious programmatic shortcomings can be scheduled into the remaining education and training.

THE SYSTEM

The essential ingredients that go into the computer system are the taxonomic description of potential experiences likely to be faced by a physician's assistant. This grouping is based on the major subdivisions by body organ, body system and historic groupings. (Ref. 2) Figure 1 lists the major groupings involved, while Figure 2 develops skin problems at a finer level. In addition to reporting their individual patient contact by one or more of the four digit categories described above, the PA also lists the level at which each clinical experience was viewed. Figure 3 displays the card currently used as input to the computing program involved in the system.

Each of the levels inpute a larger and larger involvement of the PA in the treatment process. Figure 4 displays the verbal descriptions of the responsibility level definitions. As can be seen from Figure 3, the PA is also asked to check the location of the visitation as well as the age of the patient. Information about drug prescribing is also collected.

THE COMPUTER ENVIRONMENT

The computer chosen to process this data was an IBM 5110 computer with 48K bytes of storage and two one million byte floppy disk drives for auxiliary storage. The computer can be programmed in BASIC or APL and is capable of communicating with large computer systems through a communications adapter. The system as configured also has an attached matrix printer.

Major Classifications of Medical Problem Areas

- | | |
|----|-----------------------|
| 00 | MISCELLANEOUS |
| 01 | SKIN |
| 02 | CARDIOVASCULAR |
| 03 | PULMONARY |
| 04 | MUSCULO SKELETAL |
| 05 | H.E.E.N.T. |
| 06 | ENDOCRINE |
| 07 | EMOTIONAL PSYCHIATRIC |
| 08 | OB/GYN |
| 09 | GASTRO- INTESTINAL |
| 10 | GENITO-URINARY |
| 11 | INFECTIOUS DISEASES |
| 12 | HEMOPOETIC |
| 13 | NEUROLOGIC |
| 14 | PROCEDURES |

Figure 1

- | | |
|------|------------------------|
| 01 | SKIN |
| 0101 | Dermatitis, Contact |
| 0102 | Cellulitis |
| 0103 | Rash, etiology unknown |
| 0104 | Warts |
| 0105 | Burns |
| 0124 | Impetigo |

Figure 2

Definition of Responsibility Levels

- Level 1: Problems referred immediately to the physician either at the time the patient presents his chief complaints, or after limited preliminary historical data has been gathered.
- Level 2: Problems referred to the physician after the basic history, physical examination, and ordering of laboratory tests.
- Level 3: Problems for which the PA would propose a protocol or plan for management and therapy, but which would require mandatory consultation with the physician prior to the implementation of the plan.
- Level 4: Problems for which the PA would select and implement a protocol for plan for management and therapy without mandatory consultation with the physician

Figure 4

A		B		C		D		E		F		G		H		I		J		K		L		M		N		O		P		Q		R		S		T		U		V		W		X		Y		Z	
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50		
A		B		C		D		E		F		G		H		I		J		K		L		M		N		O		P		Q		R		S		T		U		V		W		X		Y		Z	
A		B		C		D		E		F		G		H		I		J		K		L		M		N		O		P		Q		R		S		T		U		V		W		X		Y		Z	
A		B		C		D		E		F		G		H		I		J		K		L		M		N		O		P		Q		R		S		T		U		V		W		X		Y		Z	
A		B		C		D		E		F		G		H		I		J		K		L		M		N		O		P		Q		R		S		T		U		V		W		X		Y		Z	
A		B		C		D		E		F		G		H		I		J		K		L		M		N		O		P		Q		R		S		T		U		V		W		X		Y		Z	
A		B		C		D		E		F		G		H		I		J		K		L		M		N		O		P		Q		R		S		T		U		V		W		X		Y		Z	
A		B		C		D		E		F		G		H		I		J		K		L		M		N		O		P		Q		R		S		T		U		V		W		X		Y		Z	
A		B		C		D		E		F		G		H		I		J		K		L		M		N		O		P		Q		R		S		T		U		V		W		X		Y		Z	
A		B		C		D		E		F		G		H		I		J		K		L		M		N		O		P		Q		R		S		T		U		V		W		X		Y		Z	
A		B		C		D		E		F		G		H		I		J		K		L		M		N		O		P		Q		R		S		T		U		V		W		X		Y		Z	
A		B		C		D		E		F		G		H		I		J		K		L		M		N		O		P		Q		R		S		T		U		V		W		X		Y		Z	
A		B		C		D		E		F		G		H		I		J		K		L		M		N		O		P		Q		R		S		T		U		V		W		X		Y		Z	
A		B		C		D		E		F		G		H		I		J		K		L		M		N		O		P		Q		R		S		T		U											

Figure 3

The data entry process is limited by loading the program disk which through the use of the APL "latent expression ($/\underline{\quad}/\text{LX}$)" automatically causes the system to present the data entry clerk with a menu of options. (See Figure 5)

Though each of the options are relatively speaking self-explanatory, some supervisory orientation is necessary to train a new clerk. From the initial menus, templates are available to perform the function chosen in the menu. The template that permits entering a new student into the system is displayed in Figure 6.

MENU

OPTIONS:

- ```

1 ADD NEW PHYSICIANS
2 ADD NEW STUDENTS
3 INPUT STATISTICS FOR
 THE STUDENTS
4 LIST ALL PHYSICIAN
 CODE #'S
5 LIST ALL STUDENT CODE
 #'S ENTERED
6 CHANGE A STUDENT'S
 ADDRESS OR DELETE
 A STUDENT
7 CHANGE A PHYSICIAN'S
 ADDRESS
8 STOP

```

Figure 5

## ENTER STUDENT CODE

2

NAME: John Student

ADDRESS: 330 St.

CITY: Lincoln, Nebr.

Figure 6

A sample of an output report is displayed in Figure 7. The system is called PAPER, Physician's Assistant Program Evaluation Report.

[illegible]

Figure 7

The two authors from the computing environment believe that APL was the best language for this project because it both permits rapid implementation of system features as well as permitting relatively rapid program modification. The authors also believed that APL programs can be developed in a remotely accessible large computer environment to run on a small computer production facility. This was not the way the system was implemented because of the relative unfamiliarity of the authors with the chosen hardware. The IBM 5110 was first chosen because it was the only APL stand-alone processor easily available with the appropriate core, tape, printing and speed requirements. The IBM 5110 was announced midway in system's implementation and possessed such features as increased speed, controlled input to and from any location on the CRT face, upper and lower case letters and random access disk capability. The communication feature was important if the remote program development concept was to be tested.

The designers and users of the system were pleased and impressed by the system's implementation quality, cost and timing. (Ref. 3). Now that the system is in production, the system's designers realize that the choice of APL may place some limitations upon further system development because of the relative shortage of good APL programmers. They have added the BASIC language feature to the hardware.

#### FUTURE DEVELOPMENTS

The PA program administrators desire to develop and add a second major system to their environment. They wish to build a CAI-like system similar to that described in "A Profile Examination System for Physician Extenders and a Method for Curriculum Description." (Ref. 4) They are currently evaluating a CAI package based on work by Surkan and Evans. (Ref. 5 and 6)

In the not too distant future communications capabilities with a large central processor will be tested so as to examine the concept of remote program development as well as large scale statistical analysis.

#### References:

- Ref. 1 Ekwel, R., MD, Gallagher, T.F., MD, Edwards, J., MS, Frary, T, P.A.C.  
"The Physician's Assistant: An Occupational Analysis", 1976. University of Nebraska Medical Center
- Ref. 2 Valgora, V., "Taxonomy of Medical Problems Seen by a Physician's Assistant", 1976, University of Nebraska Medical Center
- Ref. 3 Sundemeier, C. (Ed.), "Computer Augmented Instruction/ Experimental Program: A First Year Report", 1979, University of Nebraska Computer Network
- Ref. 4 Small, N.W., MD, "A Profile Examination System for Physician Extenders and a Method for Curricular Prescription", Final Report 1977, University of Missouri-Kansas City School of Medicine
- Ref. 5 Surkan, A., Evans, R., "Computer Delivered Testing", Computer Aided Test Construction Digest, Educational Testing Service, Princeton, New Jersey, Vol 2, No. 1, 1977
- Ref. 6 Surkan, A., Evans, R., "APL Package for Personalized Courses with Testing, Scoring and Record Keeping", Proceedings of the Symposium on Personal Computing National Computer Conference, Anaheim, Calif., 1978

**COMPUTER RESOURCES:**

**HARDWARE**

**SOFTWARE**

**PEOPLE**

**FUTURES**

# CAMIL AND THE EFFECTIVE IMPLEMENTATION OF INSTRUCTIONAL SOFTWARE FOR A LARGE SCALE CMI/CAI SYSTEM

by

David Pflasterer  
and

Anne Montgomery  
McDonnell Douglas Corporation

The approach taken in developing instructional software for the Air Force Advanced Instructional System (AIS) is described. The AIS is a large scale computerbased instructional system designed to improve efficiency of technical training for up to 2000 students.

The Computer Assisted/Managed Instructional Language (CAMIL) was designed to facilitate instructional software development. It will be shown that this language is more effective for large scale Computer-Managed and Computer-Assisted instructional systems than languages such as TUTOR, PLANIT, COURSEWRITER, TAL, and NATAL. In addition, human-engineered software for defining data and decision rules for Computer-Managed Instruction and an easy-entry authoring system for Computer-Assisted Instruction will be discussed.

## 1.0 Introduction

The Advanced Instructional System (AIS) is a prototype computer-based training system developed to demonstrate the feasibility of administering and managing individualized instruction on a large scale. The primary operational function of the AIS is to provide training and management for up to 2000 students per day in three selected courses taught at Lowry Air Force Base, Colorado. A secondary but major function of the AIS is to provide a research and development capability for evaluation of instructional innovations. In order to fulfill these functions, the AIS has incorporated state-of-the-art instructional techniques, media, computer hardware and software.

The software, including both Computer-Managed Instruction (CMI) and Computer-Assisted Instruction (CAI) capabilities, provides a versatile automated system for preparing and presenting instructional material as well as for evaluating student progress and course quality. To provide the system level software support, the Computer Assisted/Managed Instructional Language (CAMIL) has been developed in addition to a time-shared operating system and data base management facility.

## 2.0 The Computer Assisted/Managed Instructional Language (CAMIL)

### 2.1 The Requirements of the Language

During the specification of requirements for the language, certain priorities were established. The top-level purpose of CAMIL was to provide a wide range of users with an efficient means of interacting with the computer to perform large scale (up to 2000

students) CMI and CAI.

The user community was to include application programmers who would develop programs to model data and manage Air Force training in a more efficient way (i.e., CMI); instructional authors and planners who would create CAI materials; and system programmers utilizing the language to develop functions such as source program editing and file management.

In order to accommodate such a wide range of users, a general purpose language was designed. Language constructs were created to accommodate 1) the development of CMI programs; 2) the construction of CAI materials; and 3) the implementation of system programs.

The language is currently operational as the primary programming language of the AIS. The features it provides for CMI and CAI are discussed in Sections 2.2 and 2.3 below. The approach to provide "ease of use" for its users is discussed in Section 2.4. And, Section 2.5 compares CAMIL with instructional languages currently in existence.

### 2.2 The CMI Facilities

In order to support CMI, a full range of standard data types is provided, including INTEGER (and subranged integer), NUMBER, LOGICAL, CHARACTER, STRING, TEXT, CLASS, and SET. The data structuring methods available include ARRAY and RECORD types. New data types, consisting of groups of the standard data types, can be defined by the user via the TYPE statement. The general "flavor" of the procedural component of CAMIL is similar to existing ALGOL-based languages. Procedures are easily defined; the scope of variables can be controlled through



the use of blocks; and statements can be combined into logically meaningful groups through the use of compound statements.

CMI constructs found in CAMIL are the IF THEN ELSE, CASE, iterative, procedure call, RETURN and GOTO statements, and the IF expression.

Creation, access, and modification of large data base files is also a part of the CMI facilities of the language. This large scale I/O facility is necessary to effectively manage the large numbers of students active within the CMI system.

### 2.3 The CAI facilities

In presenting the CAI facilities within CAMIL, two levels of capabilities should be discussed. First, general capabilities (found in both instructional and non-instructional languages) required for CAI, are covered (e.g., interactivity, graphics, and data base I/O). And secondly, capabilities unique to CAMIL which were designed to provide power beyond the general capabilities are described (i.e., features which allow creation of powerful CAI statements that are unique to a particular area of instruction). Sections 2.3.1 and 2.3.2 below describe the two levels of capabilities.

#### 2.3.1 General CAI Capabilities

Section 3.4 describes the approach taken in the AIS for creation and presentation of CAI lessons. In this approach the author does not have to know computer programming or relate instructional material to an individual who knows computer programming. The language features discussed in this section are those which are (and were) effective in the development of the software which implements the authoring and lesson presentation system.

The CAMIL language facilities usable for CMI also apply toward the effective creation and presentation of CAI materials. IF THEN ELSE, CASE, iterative, procedure call, RETURN, GOTO, and data base I/O statements are all used to efficiently and reliably develop the software of the CAI system.

Additional CAI features, not applicable to CMI programming, are also available within the language. These features are those which support author/student interaction with the computer via a video display terminal, presentation of graphics, student response analysis, and author/student interruption of authoring/instructional sessions.

Interactiveness between author/student and the computer is necessary in a CAI environment. CAMIL statements such as

```
Erase Screen;
Write 'Enter a response of 20 characters
or less';
Accept Into answer At Line 50, Col 1; {wait
for a student response}
```

provide the necessary interaction within the CAI system.

The presentation of graphics is accomplished with statements such as

```
Draw From x_axis1, y_axis1 To x_axis2,
y_axis2; {draw a line segment}
Draw Circle Radius 3 Inches at x_pos, y_pos
From 0 Degrees to 180 Degrees;
{draw a 180 degree circle of radius 3
inches}
```

Student responses can be compared with anticipated responses with a JUDGE statement. An example is:

```
JUDGE
BEGIN {ask question and wait for student
response}
Write 'What is one important unit within
TRANSDUCER-35 ?';
Accept; {wait for a student response}
END {ask question and wait for student
response}
WITH BEGIN {have student response}
'CIRCUIT BREAKER-25',
'TRANSISTOR-140',
'DIODE-46': : Write 'Good Show!
Proceed to Next
Frame';
'DIODE-47' : Write 'Not quite.
DIODE-47 is in
TRANSDUCER-36';
END {have student response}
ELSE Write 'No. Maybe you misunderstood
the question. Let's try again.';
```

Upon entry into the JUDGE statement in this example, the student is prompted at his terminal to respond to the question 'What is one important unit within TRANSDUCER-35 ?'. Upon receipt of a response, the various anticipated responses are examined and if a match occurs on any one of them, the action associated with that anticipated response is executed and the JUDGE statement is exited. If the response fails to match any of the anticipated responses, the ELSE portion of the JUDGE statement is executed which displays 'No. Maybe you misunderstood the question. Let's try again.'. An automatic branch then occurs, taking the program up to try the question again. Counters could be placed in the ELSE portion of the JUDGE statement thereby keeping track of the number of wrong answers and not allowing an endless loop of wrong answers.

Interruptions to authoring and instructional sessions are provided with ON statements. An example is:

```
ON BACK DO
BEGIN {user has depressed the BACK key}
exit ← TRUE;
RETURN;
END;
```

In this example when the BACK key is pressed, the "exit" flag is set and the module is exited.

### 2.3.2 The Language Extensibility

An advanced author desiring to create instructional materials without utilizing the authoring system may do so via the extensibility features of CAMIL. The language has been designed to allow natural-language-like statements to be easily added for particular areas of instruction. If, for example, the application area is pilot training on an F18 fighter, sentences could be developed by programming personnel for use by authors. An example is:

```
Display F18 Instrument Panel;
Set Altitude To 5000 Feet;
Set Air Speed To 800 Knots;
JUDGE Input WITH
BEGIN
 STICK : (Adjust Direction; If Error Then
 Display);
 THROTTLE : (Adjust Air Speed; If Error
 Then Display);
 LAND_GEAR : (Lower Landing Gear; If Error
 Then Display);
END;
```

A pilot could be given this lesson either at a video display or in an F18 simulator. Once the F18 panel is established in this example, pilot inputs could be received. The pilot specified actions would be performed and checks would be made to determine if pilot errors had occurred. If so, immediate feedback would be given to the pilot.

The language extensibility feature has been used within the AIS but not within the area of CAI, i.e., there have been no authors deviating from the authoring system to create CAI lessons such as the one exemplified above. Should advanced authors desire to use this capability in the future, it is available.

### 2.4 The System Approach

Rather than just a language, CAMIL can also be thought of as a system made up of a language, compiler, executor, loader, and source program editor. There is no need for the user to manipulate source, object, and executable load modules. The only concern is with creating, updating, and compiling the source program via the source program editor. The object code for a program is automatically stored at the end of the compilation and loaded into computer memory when the user specifies the execution of the program. More detailed information on the CAMIL system approach can be found in (Pflasterer and Stacey, 1977) and (Pflasterer 1978).

### 2.5 CAMIL as Compared to Other Instructional Languages

Notice that the CAMIL constructs in Section 2.2 above for CMI programming are similar to constructs found in other non-instructional languages such as PASCAL, ALGOL-60, and PL/I. This fact, coupled with the support of CAI (given in Section 2.3 above) within the language is what makes CAMIL more effective than existing instructional languages for large scale CMI/CAI instruc-

tional systems. Examination of available documentation for TUTOR (Avner, 1973; Sherwood, 1973), PLANIT (Butler and Frye, 1970), COURSEWRITER (IBM, 1974), TAL (Fine, 1978), and NATAL (Westrom, 1977), shows that while these languages are strong in the area of CAI (e.g., interactivity, graphics, and response analysis), they do not possess the necessary constructs to support large scale CMI. To effectively manage large numbers of students (e.g., 2000 or more), many student records, hierarchy records, lesson characteristic records, test files, and resources must be efficiently maintained (see Sections 3.1 and 3.2 below). Therefore, capabilities are necessary within the language for 1) large scale, efficient, and reliable data base I/O (i.e., sequential, indexed sequential, and direct access files; file sharing; and record reservation); 2) efficient determination of correct paths (from a large number of available paths) for proper student management (i.e., CASE statements, IF THEN ELSE statements, and IF expressions); 3) extensive iteration to perform processes which are redundant for large numbers of students (i.e., a powerful and flexible iterative statement); and 4) typing of student and management data (i.e., TYPE statements). CAMIL is more powerful in these areas than are existing instructional languages and CAMIL is as effective as existing languages in the area of CAI.

It may be asked why it is necessary that both CAI and CMI software be implemented in the same language. Certainly a language such as TUTOR could be used to implement CAI software and a language such as PASCAL could be used to implement CMI software. The problem here is one of sharing instructional software between different computer installations. Many special purpose instructional languages have been created, causing concern that instructional software cannot be easily exchanged (Brahan, 1973; Lower, 1973; Paloian, 1974). With one general-purpose language for both areas of instruction, only one compiler is necessary to transport the instructional software to another computer system. There is no need to be concerned with several compilers which may or may not be implemented on a desirable target machine.

To implement an instructional system whose goals include large scale CMI/CAI and transportability of instructional software, a general purpose language, such as CAMIL, is recommended. A more detailed examination of the CAMIL language can be found in (Pflasterer, 1978).

### 3.0 CMI/CAI Instructional Software

The AIS is currently the foremost example of a comprehensive computer-based instructional system in which CAI is embedded in the context of CMI. Such an integrated system has a number of distinct advantages. Student pacing being individualized, is compatible with efficient use of CAI. Student performance on CAI lessons can be recorded directly by the CMI system. Most importantly, the extensive student performance records maintained as part of CMI can be readily accessed to provide truly individualized CAI,

280 when and where it is most needed. Thus CAI is seen as providing one of two or more alternative treatments (modules) for teaching a lesson and, as such, CAI materials are packaged and assigned as modules by the CMI system. Student terminals required by CAI are treated as instructional resources managed and assigned by the CMI system.

The remainder of this paper discusses the CMI and CAI systems in greater detail.

### 3.1 CMI System Overview

The CMI system manages the student's instructional process in a manner that minimizes contention between the order in which the student must take lessons, the best lesson treatment for the student's abilities, and the types and quantity of resources required. The portion of the CMI system that performs these functions includes:

- 1) A leveling function that determines where in the course a student should be;
- 2) A lesson finder that determines which lessons can be taught next;
- 3) An adapter that does individualization; if there is more than one way that a lesson can be taught, the adapter produces a relative weight for each alternative treatment. There are several rules that the adapter can use to determine the best lesson alternative. Since the CMI system collects a substantial amount of data describing the student and his within-course performance, this data may be used in regression and heuristic equations to predict the best alternative lesson.
- 4) A resource allocator that determines the preferred way to use the resources;
- 5) A scheduler that makes student assignments by balancing individualization and resource allocation; for each lesson treatment being considered, the scheduler computes a weighted average using the adapter's weight and the resource allocator's weight. The treatment with the highest weighted average is assigned.
- 6) A prescription handler that prints the student's next assignment and the results (if any) of the current transaction.

### 3.2 CMI System Data Base Editors

While the AIS CMI functions are supported by the CAMIL software just described, the day-to-day operation of the CMI system is controlled through a set of data base editors. The intent of these editors is to allow course personnel with no programming skills to define the characteristics of their course and establish the rules by which student assignments are made.

The implementation approach for CMI was that of a totally "table driven" system. This means that the logic which is unique to each component is represented in the form of data

stored on the data base, rather than specific program code. At each point where an instructional decision is required, an attempt was made to allow for a variety of decision rules. The indication of which rule was to be used (and in some cases the rule itself) was then treated as a data item. As an example, the regression equations for predicting time and score on a module are implemented as a generalized equation evaluation routine which obtains its parameters from the data base.

While more difficult to implement, this approach has several distinct advantages in terms of maintainability and usability. Software changes are only required when the basic operating philosophy of the system is altered. Normal operational changes in course configuration, resource inventories, and student assignment selection rules are made by changing the course data base. Since course changes do not require program modification, manpower requirements are less. Air Force personnel can maintain and administer course content and changes without programming support.

There are approximately 50 data base editors supporting everything from course structure to system security. The basic editor design is that of a template program and a set of support routines that perform the interactive query functions. File specific modifications are made to the basic template program to produce a data base editor for the file. Since basic program control is established in the template program, coding changes to produce new data base editors are cookbook in nature and can be performed by entry level personnel. Since all data base editors are offshoots of the same template program, a standard query format is maintained.

The template program approach has worked quite well in terms of software support requirements. Editors are quite easy to maintain. A new editor can be created and checked out in an afternoon. Because all support routines are common, they can be maintained in a central program library. Programmer manpower requirements are minimal. An entry level programmer can perform the maintenance and development functions.

Operational experience with this approach has also been quite positive. Despite the complexity of the data base, course personnel have been able to use the editors to institute data base changes appropriate to their needs. This is not to say, however, that the use of the editors is totally simple and straightforward. The data base and the interaction between its components is complex. In some cases, this complexity was not adequately taken into consideration in the human engineering of the editors.

### 3.3 CMI Data Analysis System

A major function of the CMI system is to provide an R & D tool for the evaluation of instructional innovations. To this end, a comprehensive data analysis system has been



developed.

Historical performance data is periodically collected and stored on disk or tape files for use by report generation programs and the data retrieval system.

Two report programs are provided to generate statistical summaries of 1) student and instructional system performance data (times and scores) and 2) test performance data at various levels ranging from total performance to single item analysis.

An information retrieval system is also available to provide the interface between the AIS data base and standard statistical analysis packages in those cases where more detailed reporting and analysis are required than those provided by the standard AIS reports.

The data analysis system has provided a useful tool for course evaluation purposes. In addition, the comprehensive historical performance records have provided a data source for R & D activities occurring both in the military and in several universities.

### 3.4 CAI System

A CAI authoring system was developed and integrated into the AIS CMI system during the past year. Extensive analyses were done to insure that the CAI system developed would suit the needs of the projected user community. These analyses suggested that it was preferable to avoid author use of the CAMIL programming language. However, the CAI aspects of the CAMIL language and experience gained with the AIS interactive editors provided the foundation for an authoring editor. The design goals were to provide authors with a variety of options within a structured framework, to appropriately guide and prompt authors' decisions to minimize formal training, to automatically accomplish as much of the programming detail as possible, and to eliminate any need for authors to even be aware of a computer language. Prior Air Force CAI development experiences indicated that sophisticated authoring tools were of no value if authors were reluctant to or incapable of learning how to use them. Therefore, a conversational format was devised for the authoring editor. Difficult concepts such as branching strategies and response judging were implemented via a highly structured dialogue that guided the author on a step-by-step basis through the process, thereby making him more comfortable with the use of the more difficult features of the system.

The heart of the AIS CAI Authoring System is an authoring editor for generation of learning materials. The editor structures the author's task to minimize formal training. The editor is divided into two major sections: 1) a frame definition and content entry section and 2) a decision logic entry section. Decision logic is separated from frame content to allow the author to change presentation sequence without modifying frame content, and visa versa.

As defined by the editor, a CAI module is divided into objectives. The system is frame oriented and each objective can contain up to 100 frames.

There are several different types of expository frames which are all similar in that they simply present information to the student. The various types are differentiated primarily for management reasons. The author enters frame content in exactly the same format in which it will be seen by the student. An extensive graphics capability is available for use in frame construction.

The editor currently supports three types of question frames: multiple choice, true/false, and constructed response. These frame types provide the primary means of evaluating students' understanding of the materials being presented. All formatting of question frames is done automatically by the editor. As structured by the editor, a multiple choice question consists of the question stem and two to five alternatives. Once the question stem and alternative have been entered, the author is prompted to specify the correct alternative or alternatives and maximum number of attempts allowed in answering the question.

A true/false question consists of only the question stem. A predefined set of positive and negative responses are used for answer judging. Only one attempt is allowed at answering the question.

A constructed response question consists of the question stem and one to nine anticipated response categories. Definition of the anticipated responses takes the form of a highly structured dialog which guides the author through the process of defining the response judging parameters on the step-by-step basis. This approach not only simplifies the process itself but helps the author understand the critical aspects of response judging.

To provide the student with substantive information about his response, provision has been made for author entry of feedback and prompt statements. The intent of feedback is to tell the student that his answer was correct or if it was incorrect why it was wrong. The intent of a prompt message is to guide the student toward the correct answer.

In preparing a CAI lesson, definition of effective presentation logic may require the greatest thought and be the most difficult part of the process for an author to understand. The effectiveness of a lesson, however, can hinge on how well this feature is used. Therefore, particular attention was given to facilitating this aspect of the authoring process. Logic is entered via a highly structured dialog which guides the author through the process on a step-by-step basis. The result is then displayed in English. By structuring the logic entry process, the author can develop relatively sophisticated strategies without formal knowledge in computer programming.



While the authoring editor provides the vehicle by which CAI materials are developed, there must also be software to support presentation of these materials to students. CAI presentation is accomplished through the use of a general program structure and set of support routines driven by records created by an author using the editor. To alleviate the need for computer programming, a flexible template CAI presentation program was developed which would support a broad class of CAI lessons. While the services of a computer programmer are still required to expand the program to cover new applications, the extent of these requirements is reduced in two ways. First, the template program was designed to be as flexible as possible without being unduly complex. Second, given the characteristics of the CAMIL language and a well-structured documented template program, a relatively inexperienced programmer can modify the basic program to meet the requirements of new applications.

New applications requiring the creation of an additional CAI presentation program arise when authoring strategies are applicable across a large number of lessons. Although authors can define these strategies via decision logic in the authoring editor, it is generally more efficient to create another presentation program which incorporates the instructional strategies as part of the program code. Depending on the number of lessons affected, a minimal programming effort can save substantial authoring time and produce more efficient CAI lessons.

#### 4.0 Conclusion

The implementation of the AIS, as described in this paper, has produced several basic conclusions. They are reiterated in the following paragraphs.

For implementation of large scale CMI/CAI systems, a general purpose language, such as CAMIL, is most suitable. The varied programming requirements of a large scale CMI/CAI system should not be fulfilled with a language oriented only to CMI or only to CAI, but to both CMI and CAI.

The AIS has successfully improved the efficiency of technical training by managing the student instructional process. With the CMI system, the order in which the student must take lessons, the best lesson treatment for the student's abilities, and the types and quantities of resources required are all considered in minimizing the student's path through a course. Through the use of data base editors, course personnel can define the characteristics of their course and establish the rules in which student assignments are made.

Within a military environment, CAI is most effectively accomplished with an authoring system. The authoring task is simplified substantially by 1) not requiring the author to use a computer programming language; 2) making the authoring task a highly structured but flexible process; and 3) providing an easy-to-use system via human-

engineered software.

#### 5.0 Acknowledgements

We would like to thank Wilson A. Judd, Thomas G. Klem and Hadley V. Stacey for their helpful comments toward the construction of this paper. We would like to thank the Air Force Human Resources Laboratory at Lowry Air Force Base, Colorado, for funding the development of the Advanced Instructional System.

#### 6.0 References

- Avner, E., et. al., "AIDS", PLATO Services Organization, Computer-Based Education Research Laboratory, Urbana, Ill., Aug., 1973.
- Brahan, J. W., "Language Standards for Computer-Aided Learning", Int. J. Man-Machine Studies, 5, 337-345, 1973.
- Butler, A. K., and Frye, C. H., "PLANIT Language Reference Manual", System Development Corporation, TM-L-4422/002/01, Oct., 1970.
- Fine, S. "TAL User's Manual", Hazeltine Corporation, McLean, Va., 1978.
- IBM, "Coursewriter III Version 3 Author's Guide" SH20-1009-2, Oct., 1974.
- Lower, S. K., "Making Computer-Aided Instruction Make a Difference in College Teaching", Int. J. Man-Machine Studies, 5, 321-328, 1973.
- Poloian, A. Y., "An Interrogative Authoring System", Int. J. Man-Machine Studies, 6, 421-444, 1974.
- Sherwood, B. "Writing Computer-Based Lessons in TUTOR", PLATO Services Organization, Computer-Based Education Research Laboratory, Urbana, Ill., Aug., 1973.
- Westrom, M., "Summary and Current Status of NATAL-74", AEDS Journal, Spring, 1977.
- Montgomery, A. D., and Judd, W. A., "Development of Authoring Support Software for the Technical Training Environment", AFHRL Technical Report, Lowry AFB, CO: Technical Training Division, Air Force Human Resources Laboratory, in preparation.
- Pflasterer, D. C., and Stacey, H. V., "The CAMIL Programming System", Report Number AIS-3.8-1903, The Air Force Human Resources Laboratory, Dec., 1977.
- Pflasterer, D. C., "An Approach to Efficient, Fault Tolerant Programming", SIGPLAN Notices, Vol. 13, No. 2, Feb., 1978.
- Pflasterer, D. C., "The CAMIL Programming Language", SIGPLAN Notices, Vol. 13, No. 11, Nov., 1978.

# HUMANISTIC TECHNOLOGY: TOWARD A RENAISSANCE FOR EDUCATION

D. Thomas King  
St. Paul Public Schools  
St. Paul, Mn. 55102

The purpose of this paper is to evoke among the educational community a commitment to begin a needed renaissance in education. Education, as we know it today, has not changed substantively in centuries. Through these "darkened ages", we have failed to utilize our advancement in learning theories, to make valid assessments of needed skills and understandings within our populace, to capitalize upon our remarkable technologies so that the race might be won "between education and catastrophe." This paper suggests some guidelines as to how this symbiosis might begin.

Without discussion and some consensus within the entire educational community, there can be no progress, no renaissance. Presented here are the author's personal observations based on his experience with public and private educational systems as well as his thoughts concerning the promise and reality of technology.

It is hoped that this presentation will help stimulate the needed changes in education: education for each student which is more than educational opportunity for all. As costs rise and revenues decline, it is more imperative than ever that we examine these hopes and adopt those realities which a humanistic technology can provide. The synergy of man and machine focussed together can rekindle the lamp of learning with the brilliance of a second renaissance. Renaissance Man is indeed the "Key Component!"

"And what is good, Phaedrus  
And what is not good--  
Need we ask anyone to tell  
us these things?"

- Robert Pirsig

Among the Greek classics which have transcended time is: "The gods help those who help themselves." That Aesopian conclusion emanated from the fable in which mortal man despairingly cries for divine intervention to his many earthly problems. What Zeus was trying to tell us is to get our own act together, so to speak, and get it on. Divine blessings, resurgence and achievement are bestowed upon those who already have committed a personal, sizable investment of their own.

The antithesis of this self-reliant perspective is a mind-set which is as prevalent today as it has been many times throughout the course of human culture: a satisfaction with the status quo, a frantic search to find enough Georges to let do it, a hapless "woe-is-me" which truly characterizes a true Dark Ages of the spirit. And where this Darkness flows, nothing grows. Arts, science and learning suffer: the fruits of humanistic growth which should be brought to germination atrophy instead.

Some five centuries ago, western thought was about to break free from the medieval Dark Ages. Some striking parallels between then and now exist which evokes reflection and, perhaps, a collective call for action.

The fifteenth century witnessed numerous skirmishes between feudal lords, each desirous of expanding his own holdings and widening his power. Scientific thought was held suspect, particularly as it might conflict with traditional teachings. Those who proposed new ideas which did not support accepted views had no support and, in some cases the threat of Inquisition. Although the arts had begun to encourage a new expression in medium, topically it was restricted to monolithic, time-honored themes.

A catalysis began to occur in the midst of this brewing cauldron of inwardness and discontent. The ensuing reaction among the many forces at play--economic, politic, scientific, artistic--flowered forth into a cultural evolution the western world has not seen since: the Renaissance.

Periods of growth, in an evolutionary sense, often are followed by plateaus. But the seeds of greater growth still are present, awaiting the subsequent catalytic factors which will allow them to root and flourish again. The last great Renaissance was spawned by new modes of thinking and doing, new systems of action, a new individualism which fostered creativity. There appeared, rather suddenly, what has been called the "Renaissance Man." Typified in many ways by DaVinci, Michaelangelo or Galileo, these renaissance vanguards struggled to bring their uniqueness of thought to the times. What is relevant here is the recollection of the forces which allowed this new freedom of thought and

284 expression to occur. The growth of commerce and the new capabilities of the machine and power sources helped provide the eventual release from continual toil for survival's sake. Tasks met solely through the labor of man and beast became grist for emerging machines and tools. As man's body was freed so were the challenges to his mind.

The last five hundred years have seen the technological fruits of that creative freedom. But while we have brought the forces of our inventiveness to yield a less physically demanding existence, a needed second Renaissance has not yet been achieved. This new Renaissance calls for a freeing of the mind. Since the real task of education is to liberate the mind, this second Renaissance is dependent upon the emergence of a new educational system.

Education is the only human cultural process which is truly self-sustaining. Like the mythical Ouroboros, education must sustain itself in order to nurture further growth. This requires education for each.

In our nation, we have very nearly provided education for all. What we have yet failed to accomplish is education for each. Pragmatically and technologically, it is now within our power to provide that need.

To continue the analogy by examples from classical mythology, many believe that technology has been Pandora's box. While hoped-for needs are met through the release of technology's promises, accompanying pitfalls and problems have also escaped. Nuclear energy brings fear of nuclear accident, of contaminating waste. Computers invade privacy or invite fraud. But along with the escaping curse from Pandora's box, myth has held it that a needed exorcism was released too. If one fact has been observed from technology's blessing and curse, it has been that there are no solutions without tradeoffs, no answers without additional, related questions. In short, there is no closed system which is fully and forever workable. Just as Gödel proved no mathematical/logical system is complete, non-contradictory and decidable, neither are the scientific systems predicated on those precepts. We must view our progress with peripheral awareness and farsightedness.

We now find ourselves at another juncture point in our cultural evolution where the factors are right and the timing is ripe for a second Renaissance. If the first Renaissance brought a new creativity in the arts and sciences, this second Renaissance can bring a new level of humanism. The first Renaissance and subsequent change depended upon gears, pulleys and sources of power. The modern tool mimics man's mind instead. Thus the catalytic factor this time can help evoke a greater humanism. This factor of change is the electronic computer.

Lest this assertion seem too simplistic, the term computer is stated comprehensively, to include micros, minis, maxis and related control systems. These machines have varying levels of electronic sophistication and now are as much a part of our

day-to-day existence as the wheel.

In this nation we have devised an educational system which attempts to provide education for all. Because of that premise and the way it has evolved, however, we have failed in a very real sense to provide education for each. Formal education requires a delivery system. Our current system, devised hundreds of years ago, is group-oriented and labor-intensive. Designed to instruct groups of students of the same age, it functions through the traditional orchestrations of teachers and administrators. By its very nature then, this system is incompletely effective if our goal is education for each student. To understand why this is true, one need reflect only on the original premises: groups of students, teacher, a fixed-time course of study, norm-referenced curriculum. If students stay through graduation, nearly half of them will read, write, and compute at about an eighth grade level. This is sadly so because the present system is not designed to make it other than that.

Bloom, Bruner, Carroll and others have stated and clearly shown that nearly anyone, given sufficient time and assistance, can learn almost any subject. To avoid absurdity, there are certain sophisticated or uncommon skills and understandings which are sufficiently complex to make mastery, except for those with uncommon talent and motivations, nearly impossible. In no way does this delimit, however, the overwhelming majority of what could be learned if a system existed to permit and encourage it.

The present educational system has failed in several key sectors. To make learning a successful endeavor, that is, to succeed with each student:

1. There needs to be an agreed-upon level of proficiency in the commonly-needed skills of living: reading, speaking, writing, computing, problem-solving and "making-it" in a complex society.
2. There needs to be nationally and locally agreed-upon listings and descriptions of those needed skills: which skills are essential for all, which are essential for some, which are to be enjoyed for their own sake.
3. There needs to be developed a personal, individualized education program (IEP) for each and every student. Student, parent, teacher and other resource personnel need to assist in preparing, modifying and evaluating that IEP on a continual basis. The school has met its obligation when the IEP has been achieved or terminated by mutual consent.
4. There needs to be a more sophisticated use of instructional media to accommodate differing learning rates and styles. Students must be given sufficient time, learning aids, assistance and counsel to master various areas of study.
5. There needs to be an efficient system to quickly diagnose problems and prescribe

alternative instructional remedies for each student. If ignorance is similar to disease, then the instruction must be specific to the area of obscurity with a choice of instructional treatment that is effective.

6. There needs to be a greater overall efficiency to the educational process itself so that the student is not hopelessly muddled in confusion or listlessly unchallenged for intolerable periods of time.
7. There needs to be access to lifelong learning. Formal learning need not terminate at 16, high school, college or even graduate school. Learning is too vital to the higher evolution of man's faculties to be left to adolescent acquisition alone.

Each of these needs must be met if a new Renaissance in education is to evolve. The prudent employment of humanistic technology in education can help that evolution occur.

Even a cursory familiarity with computer capabilities is sufficient to note the obvious application of that technology to the seven needs listed above. Computer-based educational systems can maintain large libraries of learning skills--objective banks with related descriptors and exemplary items. Also maintainable are individual student learning profiles--including academic history, skills mastered, learning style, aptitudes and so on. The capability currently exists with computer-based systems not only to help to manage a learning environment but to assume an appropriate portion of the instructional process itself. Students can gain by the extent to which instructional activities and resources match those identified in their IEP. As microcomputer and videodisc technologies advance, marry and spawn less costly aids, learning activities can and should occur at other than school campuses. As the old saying put it: "parents and schoolmasters exist to be grown out of."

To date instructional technology has been regarded primarily as an add-on resource: put an overhead projector in the classroom, put a videotape system in the department, add a computer terminal to the resource center. Thanks to federal and special funding this has been done frequently. In most cases, studies find "no significant difference." The so-called "gadget" fixation is an added expense to an already expensive service. Education now costs between \$1000 and \$2000 per student per year in most schools. 80-90 percent of that amount is earmarked for salaries. Thus education is a labor-intensive market. Many would like to add more teachers because a gut-level feeling tells us this would help learning. But as numbers of students decline, so do the available dollars. Class averages go up instead of down. Morale declines and learning suffers. The solution to this seemingly irresolvable problem exists. Although it may not be simple to accomplish, it is simple to state: the humanistic use of technology.

Capital-intensive techniques must be combined with judicious use of a re-deployed educational labor force and a modified schooling system.

Students learn as individuals and not as groups. All students need common, basic skills. Each student must have personal learning goals--unique to his or her needs, talents and orientation. Testing to determine these goals, devising an IEP, and implementing an instructional program is possible through humanistic technology. To bring that about, some new or modified premises are needed:

1. Students need not be with a teacher six to seven hours a day, five days a week, 180 days a year to complete a course of study. Students need to learn how to learn and accept their share of responsibility.
2. School systems must permit students to study and test until mastery is achieved. Group activities should be appropriate and not predominate. Success for each must become the norm rather than the exception. The normal curve measures quickness not ultimate achievement; hence, given time most can learn what they have chosen to learn. There must be a proper mix between individualized and group instruction.
3. All learning need not occur in a school building--the home rightfully must recapture its prominence as a place for learning activities other than homework.
4. Educational planners must return to "basics" and design an improved educational system. In short, we will have an effective educational system if technology can be used fully and properly in concert with concerned teachers--so that each student can learn.

Many in the past who would have changed that which resists change considered revolution rather than evolution. Revolution is unlikely and undesirable. A plan which could work, or lead to one that might, needs to accomplish the following:

1. Secure support from parent, teacher, community, and legislative groups for a mastery-oriented academic program.
2. Secure support from local boards, teachers and faculty to try out new programs for a sufficient period of time to allow for modification and the possibility of eventual success.
3. Secure needed support from the legislature and other accreditation agencies to permit experimentation with new approaches. Class hours, school days, school years, courses and credits have to be modified to allow for new structures of operation.
4. Secure the necessary instructional media and computer systems to meet the educational service demands inherent in such a program. Redeployment of staff in new educational service roles likely will be necessary: courseware development, counselling, instructing.



5. Develop an organizational structure which will allow this new mode of learning to occur at no additional cost. Since technology costs will likely continue to decrease, and if staff can be assigned to new educational roles, such a system is feasible economically, as it must be.

It does not require great insight to see the critical role technology plays in this new Renaissance in education. In fact, any substantive change for the better is unlikely without technology. More critical than cybernetics and gadgetry, however, is the human element. As educators, we are not only an important component in the planning of something better, we are the key component for its success. We must join our humanism with available technology. If those steps are taken, the turn of this century may witness a new brilliance: the synergy of man and machine focussed together to rekindle the "lamp of learning" with the light of a new Renaissance.

#### References:

- Block, James H. and Lorin W. Anderson, Mastery Learning and Classroom Instruction, Macmillan, 1975.
- Cremin, Lawrence A., Traditions of American Education, Basic Books Inc., 1976.
- Fantini, Mario I., "Toward A Redefinition of American Education", Educational Leadership, December, 1977.
- Loye, David, The Knowable Future, John Wiley & Sons, 1978.
- Pirsig, Robert M., Zen and the Art of Motorcycle Maintenance, Morrow, 1974.
- Ulrich, Robert (editor), Three Thousand Years of Educational Wisdom, Howard University Press, Second Edition, 1954.

# INSTRUCTIONAL AND RESEARCH CONSIDERATIONS WITH AN INTELLIGENT VIDEO DISC

Bobby R. Brown      Barbara Fowler  
Lawrence M. Stolurow      Joan M. Sustik  
University of Iowa

The following article provides a summary of a panel presentation on an intelligent videodisc system. The presentation will cover functional characteristics of such a system, general instructional applications and research areas relevant to its use, as well as special applications or use with handicapped students. A 15-minute videotape documenting the intelligent videodisc system in operation will be coordinated with the presentation.

## THE FUNCTIONAL CHARACTERISTICS OF AN INTELLIGENT VIDEO DISC SYSTEM

The computer offers recognized advantages in learning environments: maximizing individualized instruction via computer management, complex branching, self-pacing, response sensitivity, student record keeping and testing. Yet with all of these assets, Computer Based Education's (CBE) deficiency of visual display capabilities represents a significant barrier to versatility of application. Until recently, computerized instruction has relied on CRT text display, hardcopy or graphic terminals for visual display. Integrating display of prerecorded full color images with motion and/or audio capabilities has only been possible through interfacing peripheral devices such as slide and movie projectors and audio tape machines. A new technology - random access videodisc now can provide all of the capabilities of full color display, motion and sound in a single device which may be merged with computer based instruction to form an "intelligent videodisc system."

A standalone, optical scan, videodisc system consists of a videodisc player, the prerecorded videodisc and a color monitor. The Thomson CSF videodisc player employs a laser beam to scan and decode the chroma, video, and audio information which is stored in microgrooves on a thin polyvinyl disc. The disc, which resembles a transparent LP record, rotates at 1800 RPM supported by a cushion of air. The laser beam can randomly access any information on the disc in less than five seconds. Because this is an optical rather than a stylus system, it allows reading the information without physical contact, thereby eliminating wear on the disc. The disc has a per side capacity of 54,000 frames, each constituting an individual image. This per side storage is equivalent to 675 slide trays of "stills", or, if played at a normal rate continuously 1/2 hour of motion. The action of a motion segment may be frozen in

a single image on the screen, it may be stepped through frame by frame, or played at variable rates in either forward or reverse. Each frame has an identifying number which may be optionally displayed. Additionally, the disc provides two sound tracks, which offer many options. These include bilingual applications, two levels of instructional complexity or instructional materials on one track and test item on the other.

The staff of the Weeg Computing Center at the University of Iowa has developed a generic interface between a Thomson-CSF videodisc player and micro, mini or large scale computer system. This interface places all of the previously described functions under computer control. The merger of the computer and videodisc technologies creates a system which provides greater potential for improving learning through instructional technology than has been available previously. Intelligent videodisc technology provides the potential for presenting verbal, audio and text, and video information in any combination of sequences as required by the instructional program. Advantages of an intelligent videodisc system over traditional media include: random access of display materials, rapid retrieval, large storage capacity, as well as the ability to implement the instructional contingencies and record keeping capabilities of CAI.

In the near future, the system will be configured to superimpose computer generated alphanumeric text and graphics onto the TV monitor image. This will provide the ability to caption visual materials as well as visually highlight and cue as needed. Through the use of a video projection system, intelligent videodisc technology may be used in a classroom environment where, under teacher control, it can present instruction to multiple users.

The intelligent videodisc system represents not only a new product, but also a new concept in instruction. The multiple "technological qualities" (Salomon, 1978) or "structural attributes" (Heidt, 1976) of the intelligent videodisc system, described in the previous section, need to be assessed for their psychological effects, that is, the interaction of intelligent videodisc attributes and learner psychological characteristics in information processing activities, and instructional effectiveness, that is, the contribution of these attributes relative to specific learning tasks (Salomon and Clark, 1978).

Unique attributes, such as interactive rather than linear video, are present in the intelligent videodisc as is the integration of previous attributes. As such, the system uses a new "language" or "symbol system" (Broussard, 1976) (Salomon, 1978) for presenting information. And as such, we need to consider how these attributes can best be used for instruction. Additionally, since use of a medium develops skills in extracting information from that medium (Olson and Bruner, 1976), we must consider whether and how to use the intelligent videodisc system to develop these skills.

For instruction, this system can assist learning by providing what Olson and Bruner categorize as "contingent experience," "observational learning" and a "symbolic system." As represented by the intelligent videodisc system, these include: contingent - CAI, observational - film, animation, symbolic systems - print, visuals.

Wodtke (1967) identified key interface requirements between students and subject matter for computer mediated instruction. These included accepting a range of answers, fast print rate for text, response processing, audio, graphics, playback capability for repeated viewing, slow motion, stop action, prompting, assessing latency, voice detection and object manipulation. The instructional importance of an intelligent videodisc system is its ability to offer all save the last two categories, which may be better delegated to the teacher.

Specific instructional applications with the intelligent videodisc could involve selectively editing existing films using available motion displays (i.e., replay a sequence, forward or reverse, variable rates, stop action), providing information retrieval of relevant text, charts or tables via random access to a still frame, presenting still frame of visual material with computer generated text, graphics overlaid, all interwoven in convention computer based instructional formats such as tutorial, drill, simulation.

In addition to individualized learning, when interfaced to a videoprojector, the intelligent videodisc system can be exercised by the teacher in a class presentation format. This would enable the instructor to present processes and

concepts, allowing students to discuss and offer an input in an "if-then" mode. This interactive process should help students develop a feel for relationships and varying parameters.

The use of the intelligent videodisc system in a learner controlled environment appears to have great potential. Pedagogically, the process of selecting material and constructing ones own instructional strategies and sequences may develop valuable skills. Additionally, given the diversity and sophistication of potential instruction via the disc, adult learners may well be inclined to selectively exercise material in the course of pursuing self directed learning experiences.

The intelligent videodisc system is a potentially rich research environment. Empirical research in instructional programming is strengthened by the record keeping capability of the computer based component. Another area of research includes the feasibility of on-line testing with visual as well as verbal stimuli. The ability to assess the impact of attributes on the processes as well as the outcomes in a learning sequence should also be researched. This line of research should adhere to the Trait-Treatment-Task Interaction (TTT I) approach where relevant individual differences in traits are used to design treatments available with the videodisc options.

Investigation is also needed to determine appropriate instructional design models and procedures using an intelligent videodisc system. While we can look to established approaches currently in use with several large scale CAI applications, the unique features of this system require refinement of, if not an alternative to, these design models.

It may well be advisable to incorporate "direct observation and unobtrusive nonreactive measures" (Mischel, 1977) in assessing the impact of this new technology. Likewise, longitudinal research is warranted to assess the assimilation of this technology in terms of its long range impact on thought processes.

These ideas are intended only to suggest the nature and scope of research and applications seen at this early stage.

No doubt prototype instructional applications and research will lead the community of users of this technology to researchable questions and patterns of utilization not yet foreseen.

#### CONSIDERATIONS FOR THE STUDENT INTERFACE RELATING TO SPECIAL EDUCATION APPLICATION

Education of the Handicapped brings into focus the requirements for flexibility and versatility in the student interface both with respect to display requirements (visual and audio) and with respect to response. Special education is concerned with exceptional children in general and in particular those who deviate mentally, socially, and/or physically from average children



to such an extent that they require some form of special treatment and services to develop their full potential. Public Law 94-142 clearly specifies that educational instructions should provide equal opportunity for all handicapped, and the concept of the least restrictive environment is critical in this regard. Estimates vary considerably with respect to the numbers of such students and these estimates will change depending upon the nature and sophistication of the detection devices used to identify students who not only have the more conventional types of handicaps but, for example, those who also have what are called learning disabilities. Kirk (1962) estimated that 12.5% of the school population could be considered handicapped in one way or another.

#### The Deaf and Hard of Hearing

Educational provisions for children with auditory disabilities are dependent on the ability to communicate visually. Since the deaf child lives in a world of auditory language deprivation, it is difficult for him to acquire general language efficiency. Research has shown that grammar for deaf children who have established base language competencies can be improved through programmed learning (Stuckless and Birch, 1964). The intelligent videodisc offers opportunities to include and surpass interactive text in educating the deaf and hard of hearing.

For example, the development of speech is important for the deaf child if he is to communicate efficiently with hearing persons. Since the deaf child cannot effectively utilize auditory feedback in developing speech, other cues must be employed. Future applications with the intelligent videodisc could incorporate the visible speech cathode ray tube translator to transform the deaf student's response into a graphic pattern. The videodisc recordings of the expected speech pattern could be displayed and superimposed on them could be the computer representations of the speech patterns produced by the deaf student.

Speech reading, or lip-reading, is another important communication need of children with auditory disabilities. Speech reading is dependent upon the child's ability to analyze in a meaningful way various lip and facial movements pronounced in the act of talking. The child must learn to read all configurations of lips and facial expressions. An interface provided by a videodisc connected to a computer could be programmed to display representative facial types pronouncing the same or different words and phrases so as to provide the basis for discrimination learning that is so essential to the child with auditory disabilities. This kind of intensive practice opportunity with a capability of branching to visual displays that represent difficult discrimination problems in learning to lip-read, could be an important application of the intelligent videodisc. It would appear to be most feasible and effective to utilize the videodisc to provide captions as needed by students with auditory disabilities

on a discriminating basis. By this we mean that through programming the captions would be either displayed or not displayed depending upon what the student indicated when signing on to the device. If the student answered a question that indicated that he or she had an auditory disability, then materials which would normally have an auditory accompaniment for the normal student would be given the additional information so that as they are displayed the captions would appear.

#### The Visually Handicapped

Educational materials used by seeing children can be adapted for the partially-seeing child in the form of large-type printing and/or through the use of low/vision aids. Since there is no evidence to support the idea that partially-seeing children differ from seeing children in educational achievements, it can be assumed that educational programs for the seeing children could be modified, through the use of a proper display device, and be useful with the partially-seeing child.

The concept that seems worth exploring in developing the use of the intelligent videodisc for the handicapped is one in which various degrees of magnification of materials might be stored along with usual print size. Materials of instruction could be programmed in such a way that the student would immediately be able to control the size, contrast, and brightness of the image to fit his individual needs. Thus both computer generated print and video images of an iconic form stored on the disc might be represented in various degrees of magnification. In this way conventional instructional materials would receive wider usage than at present simply by enlarging them as needed by the student with visual problems. This will enable the student to read most books with a minimum of difficulty.

#### FACTORS INHIBITING THE INTELLIGENT VIDEODISC SYSTEM

A number of factors may inhibit the entry of the intelligent videodisc into the educational system and can provide possible solutions. These include, but are not limited to, system reliability, an emphasis on teacher preparation, cost, availability of courseware, size, language, research and pilot programs. (Braun, Winslow, Margolin and Mirsch, Anastasio, Alderman.)

#### REFERENCES

- Alderman, Donald L. and Mahler, William A.  
 "Faculty Acceptance of Instructional Technology: Attitudes Toward Educational Practices and Computer Assisted Instruction at Community Colleges," Programmed Learning and Educational Technology, Vol. 14 no. 1, pp. 77-91, February 1977.



290 Anastasio, Ernest J., Morgan, Judith S. Study of Factors that have Inhibited a more Wide-spread Use of Computers in the Instructional Process. EDUCOM, Interuniversity Communications Council, Inc. 1972.

Braun, Ludwig. "Microcomputers and Video Disc Systems: Magic Lamps for Educators?" Paper prepared for The National Institute of Education, Washington, D.C., April, 1978.

Broussard, G. A New Medium in Video Communications The Video Disc. Paris, France, Thomson CSF, 1976.

Heidt, Erhard. Instructional Media and the Individual Learner. Nichols Publishing Co., New York, 1978.

Kirk, Samuel. Exceptional Children. 2nd edition.

Levien, R. E. The Emerging Technology. McGraw-Hill, New York, pp. 543-61. 1972.

Margolin, Joseph B. and Mirsch, Marion R., eds. Computers in the Classroom: An Interdisciplinary View of Trends and Alternatives. New York, Spartan Books, 1970.

Mischel, Walter. "On the Future of Personality Measurement." American Psychologist, April 1977, pp. 246-254.

Olson, David and Bruner, Jerome. "Learning through Experience and Learning through Media" in Man, Media and Symbols. NSSE Yearbook 1976.

Salomon, G. "On the Future of Media Research: No More Full Acceleration in Neutral Gear." Educational Communications and Technology Journal, 1978, 26(1), pp. 37-46.

Salomon, G. and Clark, R. "Media Research Methodology." Review of Educational Research, 1977, 47(1), pp. 99-121.

Winslow, K. A. "Videodisc in your Future." Educational and Industrial T.V. May, 1975. pp. 21-22.

Wodtke, Kenneth H. "Educational requirements for a student - subject matter interface." Vol. 30, AFIPS Conference Proceedings, Thomson Book Company.

LOGICAL DISTRIBUTED DATA  
PROCESSING FOR SCHOOL DISTRICTS

S. M. Raucher, Director  
Management Information  
and  
Computer Services Department  
Montgomery County Public Schools  
Rockville, Maryland

The Montgomery County Public Schools is a large school district serving 107,000 students residing in a 500+ square-mile area. As such, the district operates and maintains a fleet of more than 1,100 vehicles, including 650 school busses, 305 maintenance and supply trucks and 105 automobiles. To operate such a fleet requires over 2.4 million gallons of fuel per year.

The subject paper describes a pilot logically distributed processing application for gas dispensing. Through the use of microcomputers, a minicomputer, and the central system, the district validates drivers and vehicles, authorizes dispensing of fuel, maintains records of gas consumption, and substantiates claims for state reimbursement of transportation costs. Using this application as a logical model, the paper generalizes this approach to distributed processing for several other school district applications.

Montgomery County Public Schools, the twentieth largest school district in the United States, services a geographic area in excess of 500 square miles. The district is composed of urban areas, situated close to the border of Washington, D.C., suburban areas and rural/agriculture areas in the up-county. The 107,000+ youngsters attending the Montgomery County Public Schools attend 189 school buildings, staffed directly and indirectly by 11,800 employees. Given the Board of Education's mandated 1.0 elementary and 1.5 secondary miles to school as maximum walking distance, it is obvious that a district of this size has a significant responsibility for the transportation of youngsters. In fact the district operates in excess of 650 school busses. In addition, the district maintains a fleet of 262 maintenance trucks, 104 passenger cars, 32 supply trucks, 11 transportation trucks, and 69 other miscellaneous vehicles. The total fleet operated and maintained by Montgomery County Public Schools thus is in excess of 1,128 vehicles. In fiscal 1978, the school system utilized 2,379,871 gallons of gasoline.

School system vehicles may "gas up" in any of five depots operated by the school system. These depots, including bus barns and maintenance shops, dispense both regular and unleaded gasoline to the school system's fleet. School system vehicles may also gas up on an emergency basis, at any of the gas dispensing stations of sister-county agencies, i.e., police or

fire departments, or in an extreme emergency by utilizing the services of a commercial filling station.

The State of Maryland reimburses local school districts for the total operating cost of transporting youngsters to and from school. As such, accurate records must be maintained by the local school district as to the cost of gas and oil consumed by vehicle, as well as such other information as driver's salary, maintenance activities, parts, tires, etc. As late as 1974-1975 school year, these records were maintained manually by clerks in the Division of Transportation utilizing large ledger books with one or more pages per vehicle for the 1,100 vehicles operated by the school system.

As early as 1969, the Montgomery County Public Schools recognized the need for an integrated schools information system. Initial planning of this system continued throughout the first/several years of the 1970's and evolved into a concept of five interrelated administrative subsystems. These subsystems specifically included a Pupil Subsystem, Personnel Subsystem, Financial Subsystem, Materials Subsystem, and Facilities Subsystem. Given resource constraints, shifting developmental priorities and other outside factors that typically inhibit large-scale development in public institutions, a phased developmental approach was begun to implement the school information system. Initial priority was given to the Personnel, Financial, and

**292** Pupil Subsystems. As such, implementation of a true integrated management information system to handle the needs of the school system's transportation problems was not immediately forthcoming. Instead, a transportation accounting application was created which would, through batch data processing techniques, accrue the data on gas and oil consumption by vehicle through a series of gas tickets filled in at the pumps by the driver. These gas tickets were then keypunched and entered into the school district's computer system for posting.

Montgomery County Public Schools currently operates an IBM 370/158 computer system. The system, utilizing OS/VS1, processes both online and batch applications. No online applications were available for the transportation portion of the Materials Subsystem. As a matter of fact, of the 180+ terminals currently attached to the computer system, 146 are devoted to instructional applications. The remaining 40 terminals are utilized both for personnel and financial information and for online programming activities by the development staff. Only in recent months, have terminals been provided to other user areas such as Educational Accountability, and the Office of Continuum Education.

The load on the computer system during prime shift hours is such that further expansion of online applications for administrative purposes, such as transportation accounting, would degrade response time for existing applications. As such, it was incumbent upon the Department of Management Information and Computer Services (DMICS) to seek a technological solution other than a direct online system for gas accounting. Given the volume of over 16,000 gas tickets per month, and the evolution of technology in the late 1970's, it became apparent that an alternative technological solution to the transportation accounting problem was necessary.

Simultaneous to the recognition of the need for a better technological solution to transportation accounting by the school system, the Montgomery County Government embarked on a project to automate data collection from their gasoline depots for county-owned vehicles. The school system, recognizing the wisdom of the county government's approach, joined with the county in the procurement process for a series of automated gas pumps/data collection stations. These pumps, marketed under the name GASBOY, are now installed in each of the Montgomery County Public Schools gasoline dispensing stations. Each pump is equipped with a microcomputer that can both authorize transactions, i.e., dispensing of gas to authorized vehicles and individuals, and record information on the

transaction. Specifically, the system works as follows:

- (1) A vehicle arrives at a dispensing point. The driver inserts a credit card that has been authorized to the vehicle into the reader in the station booth. The system records the seven-digit vehicle number which includes the agency code, vehicle ID, and a lockout character.
- (2) The system then provides to the microcomputer the four-digit date, and time stamps the transaction on a 24-hour military time--the pump number and product code of the type gasoline to be issued is then entered into the transaction.
- (3) Assuming that the credit card is valid, i.e., has not been locked out by an intervention from the central system, the driver is then instructed to key-in his current odometer mileage reading.
- (4) Gasoline is then dispensed to the required amount, or a predetermined cutoff point, and the amounts dispensed recorded in the transaction.
- (5) At this point the transaction is completed and the pump locks up.

Should the driver attempt to use a fuel that has not been authorized for that vehicle no dispensing will occur. Should more than 80 seconds elapse without the pump being turned on, the transaction will be terminated. In the event that the credit card was issued to a vehicle that was not authorized to use that pump, i.e., authorized only to use unleaded gas attempting to be utilized in a regular gas setting, an Invalid Systems ID error will occur and the system will not permit the dispensing of gasoline. Up to 300 completed transactions may be stored in the microcomputer resident within the pump at each station.

Periodically, by automatic or manual control, a minicomputer located at one dispensing station will poll the microcomputers housed in the GASBOY pumps at each location. Each pump will then be relieved of its accrued transactions and those transactions reformatted and analyzed at the minicomputer station. A Computer Automation LSI/2 (16k) minicomputer is being utilized for this purpose. The same minicomputer provides a teletype console wherein a single operator may send messages to each of the pumps on lost or stolen credit cards, valid or invalid ID's, etc., thus providing for an immediate lockout in the event of a problem situation.

Periodically, usually weekly, the minicomputer reformats all collected data for



the period of time onto a nine track tape. This tape is then forwarded to the Montgomery County Public Schools central computer facility, i.e., the 370/158. Existing transportation accounting applications may then be utilized to accept the transactions from this tape as if they had been the keypunched gas tickets mentioned previously. Reports for local and state officials are then produced by the centralized system.

It should be noted that the absence of available documentation of actual expenditure for gas and oil had previously resulted in the Montgomery County Public Schools inability to document claims for state reimbursement. This had, upon occasion, resulted in the refunding by the school system to the state of over-reimbursements not defensible by the local district.

In addition to the obvious savings to the local school district accrued by maximizing reimbursements from the state, the use of this system has resulted in the cost avoidance of two fulltime keypunch positions, several clerical positions, and in general provides a significant cost saving to the school system.

The design of the GASBOY system at Montgomery County Public Schools, may well be referred to as an approach to logical distributed data processing. While it is recognized that the definition of "distributed data processing" is fluid in today's technical environment, for purposes of this paper we shall refer to the distributed data processing environment as one wherein an intelligent controller or minicomputer is utilized to collect, refine, assimilate, or reformat data for later transmission, physical or logical, to a larger host CPU. In this sense, the minicomputer with its attendant network of gas pump based microcomputer data entry stations, fits the criteria for an intelligent controller. The production of a nine-track tape, readable and usable by the IBM 370, fulfills the requirement of the reformatting and transmission of data to a larger host CPU.

The advantages of this type of logical distribution of the data processing function are many. Most specifically, no redesign or reprogramming of significant applications that are host resident prior to moving to the distributed processing network approach, is necessary. In fact, the host CPU is unaware of the distributed processing network supporting it. This provides the additional advantage of protecting one's investment in data processing that has occurred over the previous years while utilizing the latest technology to improve the data entry/data collection function. The approach thus provides for the cost

benefit of distributed processing with the cost avoidance of reprogramming.

293

Similar approaches to the logical distribution of the data processing function are currently being utilized by the Montgomery County Public Schools in several other key areas. Amongst these is the area of instructional materials acquisition, cataloging and processing. Activity in the instructional materials data area had been limited to the acquisition of the Baker and Taylor Automated Buying System (BATAB) software in late 1976. That package provided for the automated production of purchase orders for pre-cataloged instructional materials. It also provides for the financial accounting/encumbrance of such transactions. It provides reports for the fund accounting of the instructional materials accounts by school or other ordering location.

In December 1977, the Department of Instructional Resources articulated the urgent need to incorporate curriculum codes on all future instructional materials cataloging activities. A request was made of DMICS for assistance in a study to evaluate application alternatives to accomplish this objective while simultaneously simplifying and/or eliminating much of the existing manual procedures surrounding the maintenance of the BATAB database. As a result of this study, a logical distributed data processing approach was launched using the Informatics Mini-Marc system (and minicomputer) as the distributed intelligence.

The Mini-Marc system provides the user, i.e., catalogers in the Department of Instructional Resources, the ability to both extract standard cataloging data from the U.S. Library of Congress MARC (MACHine Readable Cataloging) data service and to modify or add locally generated catalog information to the master file.

The Mini-Marc system utilizes a 64K Computer Automation CPU and tape drive, Advanced Electronics floppy disk drive, and Hazeltine CRT's. It is totally user oriented, with pre-programmed capability for display, edit, delete, build, and associated catalog functions.

User staff catalog instructional materials items through the use of the Mini-Marc system. Instructional Materials cataloging transactions are properly formatted by the minicomputer and output to a nine track magnetic tape which is forwarded to DMICS for entry into the host IBM/370 system. The central CPU verifies the correctness of each transaction and updates the MCPS Master Catalog file. The central system then prints catalog cards, and adds the data to the BATAB acquisition system database. It also produces an error report for



transactions found to contain erroneous data, and prints data gathering forms used to collect further curriculum code information.

Again, the concept of logical distributed data processing has been utilized for the collection of information, creation of the data entry by the user, formatting and sorting by an intelligent controller, and then transmission to the central system of colated unified data. This data is added to the database of existing central system applications.

The Montgomery County Public Schools is currently investigating other uses of logical distributed data processing. Amongst these is an investigation into the providing of computer-managed instruction through a distributed processing network. This system would utilize the concept of course assessment and delivery at terminals or other learning stations in the schools. The data from these assessments would be collected by a minicomputer or other intelligent controller, reformatted, and forwarded to the central computer for posting to the overall pupil database and the production of reports necessary by school administrators, teachers, and principals. Similarly, course materials would flow "downstream" from centralized authors to the mini's for delivery at the learning stations.

In summary then, the Montgomery County Public Schools remains committed to the utilization of the most current technology in the providing of service to its 107,000 students. The concept of logical distributed data processing has been proven feasible in the area of transportation accounting through our GASBOY system. That system is currently operational in pilot test mode. The concept of logical distributed data processing is under active implementation in the Mini-Marc/BATAB area, and we expect significant results in the short term. The use of this concept for the delivery of instructional systems and/or the computer-managed instruction process is currently under investigation. We are hopeful, that similar advantages to those cited above in the GASBOY and Mini-Marc/BATAB applications will be available based upon the extension of this concept into the instructional as well as the administrative area.

THE MICROPROCESSOR AS PENCIL  
by  
John D'Angelo  
Corporate Engineering Center  
Texas Instruments, Inc.

I. PRESENT STATUS

The Microprocessor as Pencil! Not too many years ago that indeed would have been a strange topic. However, we have progressed rapidly during the last few years. This item, the microprocessor, has become quite an ordinary thing. Recently I had the occasion to revisit Marshall McLuhan, who surprisingly enough, has been all but forgotten, even in this time of increased use of technology. Although he was not specifically referring to education, according to McLuhan,

"History is made up of three stages: (1) Preliterate or Tribal, in which men lived close together and communicated orally; (2) the Gutenberg or Individual Stage in which men communicated by the printed word and thinking was done in lineal-sequential patterns; and (3) the Neo-Tribal or Electric, in which computers, television and other electronic communication media have moved men back together ("implosion").<sup>1</sup>

It is interesting to note that in education we have scarcely moved into the Neo-Tribal electric stage while society, at least according to McLuhan, has.

Books and books with pages and pages have been written relative to the whys and wherefores of the introduction of an innovation, technological or otherwise. Therefore, I will not belabor the point. Practically every study at some point or another in the study will place the success or failure of an innovation on whether or not the practitioner, i.e., the teachers were involved and participated in the decisions made before, during and after the innovation's introduction. The key to the involvement is in the decision-making process used in deciding on

the innovation. The practitioners must be part of the process before the change is even considered, since they are the ones who will actually implement the innovations. The only way to logically accomplish this is to have a system in place for allowing teacher input into the school's decision-making process. However, that's a topic of another paper.

## II. WHAT IS A MICROPROCESSOR?

Perhaps it is appropriate at this time to define the microprocessor. In the sense we are using it, the microprocessor is the central chip used to process the data supplied. Used in this way it might also be considered a computer-on-a-chip. Therefore, this whole paper could have been entitled "The Computer as Pencil." To be more specific, we might say that the microprocessor chip consists of an extremely miniaturized, reduced version of several integrated circuits which are capable of processing large amounts of data at rapid speeds. The integrated circuits are devices which consist of several hundreds of transistors. A large computer consists of many integrated circuits, whereas the microprocessor may consist of a single chip. the trade-off for size is speed and user capacity.

Without spending too much time investigating, one can very easily say that not a day goes by that each of us does not come into contact with an item which uses the microprocessor. The first of these was the hand-held calculator. Very few homes are without one, and many homes have several. The calculator is built around the microprocessor chip. The microwave oven's microprocessor provides you with the capability of cooking at more

than one temperature, different times, or a combination of these. Cameras, TV sets, video tape recorders all have one or more such chips. Many of the newer cars have a microprocessor as part of the ignition system, the exhaust system, or the fuel system. In the latter case, its main purpose is to monitor the amount of fuel needed in relation to the particular driving conditions.

### III. MODIFICATION OF THE TEACHING/LEARNING PROCESS

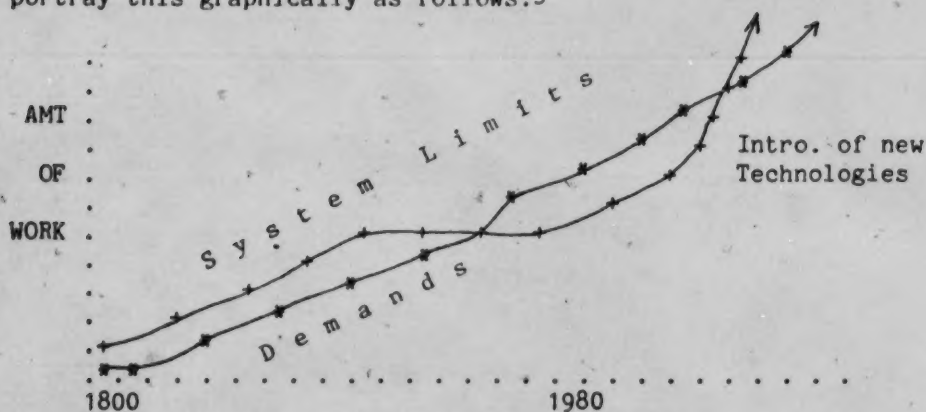
As was pointed out by McLuhan earlier, the delivery system of education has changed very little during the past 500 years. By the delivery system, we are referring to the process by which we teach, coupled to the process by which students learn. This combination becomes the process of education, i.e., the delivery system of education. With the advent of the printed word, the delivery system changed radically from the Socratic system of verbal transmission of "learning" based to a great extent on the personality of the instructor, to the more concise, formal, impersonal approach of the book. An excellent analogy is presented by Dustin Heuston of WICAT. He states,

"Educational leaders have found themselves trapped in a position similar to the one in which transportation experts might have been had they been forced to improve an old delivery system (the horse) without transitioning to new ones (the train, car, and plane). A horse can do only a limited amount of work, and experiments which attempted to improve breeding or dietary practices, or even use titanium shoes, were simply not going to produce adequate productivity gains."<sup>2</sup>

For at least the past twenty years that I've been in education, we have pumped millions of dollars into the education system of the United States, only to find that achievement scores have gone down! As our work force has increased and become much



more sophisticated, the demands on the education system have proportionally increased. As a result of this awareness, we have made many attempts to "meet the needs of our students". But, like the transportation system of 100 years ago, the present delivery system of education is no more capable of the desired level of productivity than was the horse. We have made attempts to rectify this situation. We have added language labs, hundreds of overhead projectors, and even computers to duplicate the present delivery system, all to no avail. The scores are still going down! Again allow me to reiterate, there is the very real possibility that the education delivery system has reached its maximum productivity even with all the "innovations" which have been added. We might portray this graphically as follows.<sup>3</sup>



It can be readily seen that the education delivery system has kept pace with its demand until only the past thirty or forty years. It has also been during the last thirty or forty years that we have tried desperately to make modification without changing the basic delivery system.

I must be quick to point out that my comments regarding the present education system are in no way intended to be critical of the many excellent projects and programs or of the valiant

attempts which have been made to improve the system. It is simply that the system has reached maturity. We have infused the system with much "technology" in the form of audio-visual equipment and its accompanying software, but even this has not made a significant difference. Perhaps the reason for this has been that the instructional process has continued to be serial in nature. That is to say, the student has had little opportunity to branch to an alternative point in the instructional process. It is not that we educators have not had a sincere desire to do this, but desire and good intentions have not been enough. The technology has not been available prior to the last few years.

We have even had available to us the so-called Instructional Development Process, by which we can approach instruction in an orderly, systematic fashion. This process has not met with much favor by the teachers and administrators of the education establishment. What most instructional developers have not realized is that they have inadvertently been asking to replace, or at least change the present delivery system. It was inevitable that there was resistance, for we are asking them to change all that they know and with which they feel comfortable. The Educators have learned and have been trained in the present delivery system. Anything else makes them extremely uncomfortable and rather insecure. Part of our failure, as instructional developers, and I include myself among their numbers, is that we have asked that, in order to meet the student's need, material be accessed randomly, while the technology to do so has not been available.

In their book, "Teaching Students Through Their Individual Learning Styles: A Practical Approach", Rita Dunn and Kenneth

Dunn ennumerate eighteen learning styles and nine elements of teaching style.<sup>4</sup> They, along with most traditional curriculum developers (as opposed to instructional developers) fully expect the classroom teacher to cope with these many combinations. This becomes a management nightmare. Most educators usually dismiss this idea as "pie-in-the-sky", i.e., The Impossible Dream. This dream can be summed up in two words - Individualized Instruction! For educators this has become a sacred concept, like the flag, apple pie, and motherhood.

It might well be that, like trying to improve on the horse instead of moving on to the airplane, we are trying to improve on the pencil. Rather than attempting to make pencil lead so that it will not break quite so easily, why not use something else as pencil? Then will the impossible dream become not quite so impossible? We in all our wisdom, so bent on improving that pencil, that we have lost sight of the possibility that there may be another way of accomplishing the same end. It's the old forest for the trees bit! The impossible dream is that because we have not been able to inter-relate the many teaching styles with the various learning styles. We cannot easily determine the many possible alternate paths the student could take based on his achievement at that point. Therefore, we have not been able to make available the "new" technological learning aids.

#### IV. THE NEW PENCIL

The teaching/learning process which has been in use for the past 500 years has finally reached the point whereby the results are counter-productive. The instructional tools in use prior to

the microprocessor-based learning aids tend to be group orientated with little regard to the individual. Eaton Conant from the University of Oregon has completed a study of the first four years of elementary school and discovered that the amount of time the teacher can give to an individual student is severely limited. Conant found that in a class with 25 to 30 students the average teacher was unable to devote more than 1 to 2 minutes of individualized instruction per student per day. The individualized instruction programs I have worked with all seem to become, sooner or later, self-paced rather than individualized for each student. The success of programs like TICCIT have shown a high degree of success, but at an enormous cost.

We have at our fingertips the tools which will change the delivery system. Notice I said "will" not "could", for these tools are available right now. These are instructional tools such as calculators and learning aids like the Little Professor, Wise Owl, Dataman, the Speak and Spell, and the individual, personal computer.

#### V. A SUGGESTED ALTERNATE PROCESS

Several years ago when I first read George B. Leonard's "Education and Ecstasy", I thought how nice it would be to have that kind of delivery system (although I'm sure I did not use those words since I was in the midst of the delivery system, hence I did not recognize it). In his book Leonard states,

"Every child can delight in learning. A new education is already here, thrusting up in spite of every barrier we have been able to build. Why not help it happen?"<sup>5</sup>



We thought then that was so very far into the future - a 1984 kind of future. But 1984 is only five years away and the necessary technology is here now! I would recommend reading Leonard or even reading it again. It is even more relevant now than when it was written in 1968.

But where can we go from here? We don't want to burn it all down, as they were avocating in the sixties, and start all over again (although that is a tempting thought). No, rather we must build on the present structure. The present educational establishment must be won over, for if not, the consequence might indeed be grave. What can we see in schools today? That is, not what is in schools today, but what is currently available, right now, to be placed in schools. In order to provide each student with a truly interactive environment, learning aids must be provided much as those materials (books, paper and pencils) are provided for the present passive one. These microprocessor-based learning aids should be provided, either along with or eventually, in place of their present book allotment. In some specialized areas, a simple calculator or a programmable calculator might be provided or even a speech generating device for language arts should be provided. Eventually, however, a computer is needed for each student, just as an english grammar book is needed for each student. Without considering the instructional aspects, I would guess that the first thought relative to this concept is one of cost. This is perfectly natural, especially when one is familiar with the huge cost of computer-based education of only a few years ago. But with the reduced cost of the present micro-processor based learning aids, amortized over about five years, the cost becomes a little more than one hundred dollars per year per

student. That certainly compares favorably to the cost of the student's textbook allotment per year.

The student's environment needs to have a number of characteristics which are not now provided in the present delivery system. Among these are random access to any portion of the instructional material. Presently much of the student's time is wasted while waiting for a match between his point in the instruction and the serially presented instruction. Allowing this type of interaction permits the teacher to assist the student as needed rather than to be 'on stage' all the time. Aligned with this is the concept of interactivity. The use of the individualized computer permits this kind of interaction which no teacher can presently match. The courseware of the computer is precise, patient, and always available to the student. Another attribute of the 'new' learning environment is a giant step toward achieving that impossible dream mentioned earlier, individualized instruction. We've all attested to the impossibility of the record keeping task involving 30 to 150 students each on possibly as many different programs. With the existing learning aids the record keeping chore is much simplified. We could keep track of the response and modify the student's program accordingly. The student would be branched to his level and a record kept, informing the teacher of both the student's progress and the validity of the instructional material. This permits the teacher to adjust as appropriate. Also unlike current serial media such as books or films, the courseware could be updated as the need arises. One must also remember that color, motion, and sound to which we have all become accustomed, is very much a part of this environment - and still at the before mentioned price.

But what do we need to do in order to achieve this kind of learning environment? The technology already is available. First, we must decide to do something about it; or, I suppose, we might decide not to decide. Assuming we decide to look seriously at the present educational delivery system, we must build on what already exists. We must not start with bringing in hardware-calculators or micro-computers. No, we must first start with the content of education, the curriculum itself. We must be careful not to think in terms of how does this fit into the curriculum, for the present curriculum itself must be examined. We must start with the needs of society. The jobs requiring unskilled or semi-skilled workers are rapidly diminishing, the curriculum must reflect that. A more sophisticated instructional content requires a more sophisticated delivery system.

We must involve the practitioners, the teachers, and we must have the support of the administrators. This is not a simple task, for we are dealing with attitudes. The tendency will be to cling to that which is known and with which one is comfortable. We must tread lightly but deliberately. We must define an instructional process which fits the society that presently exists. We must very judiciously look at our present resources. Rather than to seek additional monies, we might better reallocate existing monies and resources. We have pumped millions of dollars into the system to buy more of the same, i.e.: if two aides doesn't help in the classroom then let's have four; if Miss Jones does well with an overhead projector then let's get one for every teacher; if Johnny is successful using the cassette player then let's have all children use the cassette player. Unless you believe in miracles, more of the same does not promise a solution.

We have at least thirty years of data to show the extent of the improvement of our educational system using more money without changing the delivery system. It is here that the Instructional Development Process mentioned earlier can apply:

|          |                                                                                               |
|----------|-----------------------------------------------------------------------------------------------|
| DESIGN   | A process which meets the needs of students relative to our available resources;              |
| DEVELOP  | The appropriate content, produce the necessary material and acquire the needed learning aids; |
| EVALUATE | The content, the materials, and the process itself.                                           |

What if we do not change the system? Fortunately, the human species is very adaptable. But what will our society be like while this adaption takes place? More and more workers will find that they are not trained to survive. Even now we are finding business and industry needing to train their new employees in the very basic skills. In many cases, they are finding it necessary to bring them up to literacy. We have hardly begun to exploit the human potential. We know that our students are capable of achieving much more, but the scores are going down. We have at our finger tips the technology to reverse this trend. Not since the advent of the printing press have we had such an opportunity to influence the quality of life. All around us we see evidence that we have, in McLuhan's words, reached the neo-tribal or electric age, everywhere, that is, but in the world of education. It might well be that the time has come when we must stop letting schooling interfere with our children's education.

Seymour Papert of M.I.T. sums up our concept of the micro-processor in education with the following.

"When we talk about computers in education people ask how we mean to "computerize" subjects. But this is the wrong image. We do not mean to "computerize" anything. Think of the computer as like a pencil. The pencil is always present and has many uses for scribbling, for writing, for arithmetic,



for drawing. But we do not talk about "pencilizing" education. The personal computer will be there to be used in many ways like the pencil. Like the pencil, it should serve education, it should never dictate it. And it will serve it as an instrument for writing, for drawing, for doing arithmetic, for composing music and for many other purposes as well. It will be used for fun, for games as well as for deliberate learning."

.....

Note: This entire paper was composed on the Texas Instruments 990/4 computer and put into final form on a TI Word Processing computer. The computer was used as I would have used a pencil.

## RESOURCES

1. Understanding Media: the Extensions of Man, Marshall McLuhan. McGraw-Hill Book Co., 1964
2. Training, Technology, and the Educational Delivery System. Dustin Heuston. WICAT Inc. 1978
3. Ibid.
4. Teaching Students Through Their Individual Learning Styles: A Practical Approach. Rita Dunn and Kenneth Dunn. Reston, Virginia: Reston Publishing Co. 1978.
5. Education and Ecstasy, George B. Leonard. Delocurte Press, New York, N.Y. 1968.

# RANDOM ACCESS AUDIO HARDWARE, SOFTWARE AND COURSEWARE

Donald H. McClain  
Weeg Computing Center  
The University of Iowa  
Iowa City, Iowa 52242

The utilization of random access audio appears to greatly enhance many computer-based instructional learning materials and is a necessary component of others. This paper describes the features of the RA-4096 Random Access Audio device and the capabilities of computer software developed at The University of Iowa which enable an educator to employ random audio in an instructional environment. It also discusses the development of computer-based curriculum materials in the areas of music and language.

## I. INTRODUCTION

Many computer-based instructional applications would be greatly enhanced through the complementary utilization of other media; such as, audio, slides or microfiche, motion pictures, and video disks. This paper describes the RA-4096 Random Access Audio device and the computer software necessary to utilize random access audio in an instructional environment and to manage the contents of an audio disk. The audio device is manufactured by Education and Information Systems, Inc., Urbana, Illinois, and the software is written in BASIC by The University of Iowa personnel for a Hewlett-Packard 2000 computer.

## II. GENERAL DESCRIPTION OF AUDIO DEVICE

The Random Access Audio device is a tape recorder-like unit which allows individual message recordings to be accessed in a random order. Its ability to randomly select a pre-recorded audio segment is essential in an interactive computer-based lesson. To accomplish the random access capability, the device utilizes a floppy magnetic disk similar to a phonograph record except that it is constructed of a flexible material and is recorded on one side only. This disk is turned, permitting messages to be played or recorded under a magnetic head which moves toward and out from the center of disk to the various read/write locations. The disk can be rapidly turned, making possible an average access time of .6 of a second for any recording location on the disk.

The RA-4096 is controlled by the host computer and is connected between the computer and a standard ASCII terminal. The computer transmits information, under program control, to both the audio device and the terminal. In an instructional environment, an executing CAI lesson on the host computer can present to a student textual material displayed on the terminal as well as audio passages played on the audio unit.

Under program control, the student can also be given the opportunity to record his/her own vocal responses upon the disk. The instructional program executing on the host computer sends the location (address) of the particular audio segment desired to the audio device which in turn positions the magnetic record/play head to the proper place. The audio material is then either played or recorded.

The address of the recording location consists of its track and sector numbers. A disk is divided into 128 concentric tracks, referenced 0 to 127. Each track is segmented into 32 divisions called sectors. Sectors are numbered from 0 to 31 with sector 1 being the first sector on a track. (0 is the final sector on a track.) There are a total of 4096 (hence the name RA-4096) sectors on a disk, each of which can be individually addressed and accessed. Each sector contains approximately 1/3 second of audio, with a track containing almost 11 seconds of playing time. Four tracks may be accessed as a continuous message, giving approximately 43 seconds of continuous audio. The total disk can hold about 23 minutes of audio material.

In addition to the address of the audio segment, the device must be told the length of the recording, measured in seconds, in order to determine how long to run. Additionally, a bit (a variable which has a value 0 or 1) must be set to indicate whether the mode is playback or record. The computer via an executing application program transmits this data to the audio device.

The Random Access Audio device, which was designed to run with the PLATO system developed at The University of Illinois, has nonstandard data input requirements. It expects two 17 bit words of data. To utilize the device with the computer systems at The University of Iowa, an interface board was designed and built by Weeg Computing Center personnel. The interface board receives ASCII characters from the computer and decides whether the data should go to the terminal

or to the audio unit. The information for an audio request is preceded by two special control characters. If the special control characters are present, the succeeding three words of data are reformatted by the interface board and forwarded to the audio device. If the special control characters are absent, the data is switched to the terminal for display or for other terminal functions.

The audio unit was designed to utilize air pressure in controlling the read/write head movement and the spinning of the disk. Because the air requirement is minimal at 15 pounds per square inch, the air can be provided by a pressure tank or a small compressor. Initially, a compressor was utilized to fill a 120 pound tank which was attached to the audio unit through a regulator set at 15 pounds of pressure. It seemed as if the tank was always needing to be filled so alternatives were sought. Finally, a hose was tapped into the air pressure line for the building's thermostatic system. Since the air requirements are minimal, this appears to be a satisfactory approach.

### III. GENERAL DESCRIPTION OF THE AUDIO SOFTWARE

The Random Access Audio software is comprised of a series of computer programs which assist an educator in employing random access audio in his/her instructional computing applications. The audio software package provides the means of generating and/or manipulating the message location and the mode of operation parameter without the user having to undertake programming or remember message locations. The software also assists the user in creating and managing an audio disk and is designed to be easy to use with the user having to know only those features needed for his/her particular application.

A student utilizing audio in a computer-based lesson has to know nothing about the software or operation of the audio device except how to insert the disk and work the dials controlling the volume and record level. The student simply inserts the appropriate audio disk into the device and executes the specified application program. The software, in conjunction with the audio unit, does the rest by providing the student with audio output or record capability at the appropriate points as defined by the educator in the application program. If the student is to record an audio reply, he/she must wait for the red recording light to light prior to speaking into the microphone. The recording level is indicated on a built-in VU meter and is set manually by a record level control.

The course developer, on the other hand, creates and manages the audio disk and must specify the use of the audio within the application program (lesson). To aid the educator in these tasks, two basic types of software capabilities have been programmed. The first is a set of subroutines which when appended to a BASIC program facilitate the user's request for a particular audio segment. The second capability

is provided by a program casually entitled "the manager program" which assists the user in recording on an audio disk and in efficiently and easily managing the disk's contents. 309

The contents of a disk may be comprised of three different types of audio segments. The first type is a message and consists of pre-recorded audio information. Messages are of a fixed length, determined at recording times, and can only be played by the audio subroutines in an application program.

Second is the temporary response type which is used for recording a student's response during the execution of an instructional lesson. As the title implies, such a recording is only temporary and is recorded over by the next reply of the student or by the following student's session. Thus, one temporary response segment can be used over and over during a lesson. A temporary response segment can be recorded and then played back. This type of segment permits a student to record a reply and then immediately hear how it actually sounds. The recording can be compared by the student to a standard message recorded by the instructor. The comparison can be made by the student listening to both the instructor's message and his/her own audio input. This capability is extremely useful in the study of music and languages.

The last segment type is the accumulative response which is made up of a collection of student responses. It differs from the temporary response segment in that the student responses accumulate rather than being recorded over as the segment type is used. When created the accumulative response segment simply sets aside the specified amount of blank space on the disk and the application program's subroutines record the student's response in the next-available location within this area. The course developer is responsible for the management of this space via a special set of subroutines which aids in the playback or deletion of the stored student replies. An instructor can use this technique to verify how a student actually sounded on a particular test response or to collect comments during the testing stage of a computer-based lesson.

The manager program aids the user in the establishment of the three segment types. The program prompts the user for the type of audio segment desired. If a message, the program assists in the recording of the audio passage. If the segment is a temporary or accumulative response, the program formats portions of the disk to be saved for the appropriate purpose. A course developer can utilize any or all of these segment types on one disk. It should be emphasized that the user employs only the type of segment required for the particular application and needs not be concerned with other segment types. However, it should be noted too that the accumulative response can fill substantial disk space, particularly if several responses for each student are retained or if many students utilize one disk. The disk space for this segment type can be managed by the user but this arrangement requires a conscious effort. The most frequently employed audio segment will



The user of the audio software references an audio segment as a single number rather than by its track number, sector number, and length, eliminating the need for the educator to become overly involved with technical specifications. To provide this capability, the manager program establishes a BASIC formatted file for each disk which consists of a directory of used space, space reserved for responses, and free or available space. The audio segments are sequentially numbered (or the user can supply the number) as they are recorded or established, with the file containing data concerning the track number, sector number, and length of the segment. To play a message or record a response, the user simply assigns a variable in the application program to the number of the audio segment given during its creation and branches to the appropriate audio subroutine via a GOSUB statement. Messages are differentiated from responses through use of a minus sign with the number for a response segment. Utilizing this software the user has only to remember the appropriate audio segment by its assigned number.

While recording a message, the course developer should be aware of the maximum time required for the audio material. The manager program will prompt for this figure during the recording of a message and will then invoke the record subroutine so the message can be recorded. Upon completion of the recording, the user can press the return key which instructs the manager program to turn off the microphone if the maximum time has not elapsed and determine the actual length of the message. The actual time figured will be stored in the directory file together with the track and sector number. When the message is played back, the actual length will be utilized so that the message will be heard without excess black space. The address location of extra disk space, if any, will be returned to the free space list. Utilizing this technique, the course developer does not need to know the exact length of each audio message but should have a reasonable approximation for the manager program to efficiently manage the allocation of the disk space. The time required for a message is an important factor because the audio device is designed so that if a unit is serving a request for audio and another request is initiated, the current audio segment is interrupted. This situation can be useful at times but in most cases has undesirable effects. Therefore, the audio subroutines have been designed to insure that this interruption will not take place by executing a wait period until the first message or recording session has been completed.

Due to the limited total time available on a disk, a single disk will most often consist of only one application. However, a course developer might want to put more than one single short session on a disk or might wish to divide a single session into smaller components. The manager program permits a user to do this by grouping audio segments together into what are called "sets." A disk can be subdivided into from one to fifteen sets with sets being distin-

guished by unique names consisting of up to sixteen characters. The default is that the entire disk is an unnamed set. Each set can have any or all of the three audio segment types (message, temporary response, and accumulative response). The numbering system of a set's audio segments is independent of any other set; that is, there may be duplicate message numbers in two different sets. Because the numbering scheme is relative to a particular set, the user must invoke a subroutine which establishes the values for the new set every time sets are changed within an application or the manager program. So, the utilization of the set concept should be employed with the caution of not changing sets too often within a short segment of code. (It should be noted that the set concept is an advanced feature and should be used only when the application warrants it.)

The audio software has been designed to provide the user with as little or as much capability as is necessary for the particular lesson. If only the capability to play recorded passages is required, then the user must know only about messages. On the other hand, if the user has need for several different groupings of audio segments as well as need for both playing pre-recorded messages and recording student responses, then the user must have the full range of capabilities. Many applications will use features somewhere in between.

In summary, the software package for providing random access audio consists of a set of subroutines and a manager program. The subroutines which are appended to the user's program facilitate the utilization of the audio device during the instructional lesson. The user simply assigns the number of the desired audio segment to the variable and then branches the program to the appropriate subroutine via a GOSUB statement. The manager program assists the user in creating, recording, and managing the contents of a disk and establishes and maintains a directory file which contains all the information concerning the audio segments. It is not necessary for the user to be concerned with tracks and sectors as the software is designed so that he/she needs to be familiar with only those capabilities required for the specific, desired application.

#### IV. COURSEWARE

The Random Access Audio device was designed to provide brief, rapidly recoverable, audio responses for interactive applications, especially in an instructional environment. Almost every instructional application can be enhanced by the employment of audio but, of course, there are areas in which this feature is really necessary. Also, there are learner populations who, because of their physical characteristics or learning styles, clearly benefit from the addition of audio to any computer-based learning materials. Academic areas in which audio appears to be a valuable asset are music, languages, elementary school reading, spelling, speech problem identification, dialect training in theatre, nursing, medicine, and veterinary medicine. It would also appear to hold great potential for industrial and military training.

Currently at The University, computer-based learning materials in the areas of music and languages are under development. In music, a program is being created to provide practice in the identification of rhythm dictation patterns. A student can select from a group of twenty common meters the meter which he/she desires to practice. The program then provides ten rhythm patterns which are randomly selected from a pool of patterns for that particular meter. After listening to an item which is from one to three bars long, the student identifies the rhythm pattern by entering a response on the terminal's keyboard using a special music notation of letters to represent the musical notes and symbols. Each time a student's answer is required, a prompt of the musical notation appears in the right hand part of the display. Furthermore, the student has the option of having the rhythm pattern replayed and the pattern is always replayed if the student's reply does not agree with the stored answer. This program provides a method of practicing rhythm dictation that was previously possible only by having someone play the rhythm pattern on a musical instrument.

In the languages area, program development is taking place in four types of applications: sound discrimination, grammar drill, dictation, and listening comprehension. In sound discrimination, the student is asked to distinguish between pairs of sounds which are similar. He/she hears two sounds with an identification of each and immediately is given one of the sounds again. The student is then asked to identify the repeated sound.

In the grammar drill and the dictation applications, audio is employed as a prompt as well as to provide reinforcement feedback. In both cases, the student responds to questions by typing the appropriate answers. It is important in the learning of a language for students to know not only the right word or groups of words but to actually hear them spoken at the same time. The ability to have random access audio is essential for providing individualization. Many elementary language text books have supplementary audio tapes available, and it is not difficult to transfer the tapes' contents to an audio disk. The appropriate learning strategy can then be programmed, resulting in an interactive, individualized learning package.

In program development pertaining to listening comprehension, the student is given a short audio passage and is then asked questions relating to the material. The student has the option of listening to the material several times, and segments which appear to be misunderstood will also be resupplied. Developmental work in the languages of German and Spanish is being undertaken.

In summary, the employment of random access audio appears to provide a much needed dimension to computer-based education materials.

## SELECTING MICROCOMPUTERS FOR THE CLASSROOM

David B. Thomas and Donald H. McClain

The University of Iowa

© Copyright 1979. All Rights Reserved

Microcomputers suitable for classroom use are available from many manufacturers. Selecting an appropriate system requires that the educator determine his/her instructional needs and identify instructional strategies before considering a system. Comparing the characteristics of small and large computers with the use-frequency of identified strategies permits the educator to determine whether microcomputers are appropriate.

Selecting a system first requires the instructor to describe instructional requirements in terms of system capabilities. Specific systems may then be evaluated for suitability of available courseware and software, documentation, cost, service, etc. A final selection may follow which is consistent with identified instructional uses.

Microcomputers seem to be everywhere. They are found in automobiles, ovens, sewing machines, televisions, and most recently electronic games. Microcomputers may be seen in department stores, stereo shops, and computer stores. Advertisements for inexpensive systems appear in The Wall Street Journal, Sunday-editions of metropolitan newspapers, and in-flight magazines. Microcomputers were recently featured on the ABC Evening News. In short, micros are now clearly present at the consumer level and have entered our consciousness.

Teachers have discovered microcomputers, and may see micros as an inexpensive means to provide computer literacy and computer assisted instruction to students at a cost lower than that of time-sharing systems. Many have rushed to buy one of the systems which are now available; some number of these recent purchasers will find that the system they have bought will not meet their continuing (i.e., long-range) needs and will be unhappy with their systems. Others may find that the instructional needs they have are not well served by the system they have purchased.

Selecting an instructional computer system from the large number and variety of microcomputers now available can be a challenging venture even for an individual knowledgeable in instructional computing. It can be a dilemma for the non-computer administrator or teacher, given the quantity or quality of most of the information being distributed by computer vendors. A problem arises in the large discrepancy between the price of a basic system quoted by the company in its advertisement and the actual cost of a minimum system configuration capable of easily and efficiently supporting instructional computing.

The key to the successful selection of a microcomputer lies in the careful analyzing of

the instructional problem, determining of the instructional requirements, and specifying the required computing capability. Once the computing capability has been determined the potential buyer can survey the market place for systems satisfying the specified needs. The final task is selecting the microcomputer system that satisfies all the predetermined criteria.

This paper strives to guide the educator in the selection of a computer system, especially a microcomputer system, that will be used primarily to enhance the instructional process. We have attempted to outline a model which will help teachers perform the prerequisite activities necessary for the specification of a microcomputer system which will meet present and future instructional computing needs.

The importance of performing an analysis of the instructional needs and anticipated uses of the computing resources cannot be overemphasized. The assumption that a microcomputer will permit the same varieties of instructional computing as are available on existing large systems (or on mini computers) is simply incorrect. The faulty assumption is compounded when the user attempts to extend use of the micro to applications beyond those initially intended. The extensions are sometimes not feasible. The point we wish to make is that microcomputers can be valuable additions to an instructional program but that careful planning is necessary to avoid disappointment which might ensue with the selection of an inappropriate system.

## THE MICROCOMPUTER SELECTION MODEL

All too often, one begins with the purchase of a computer/terminal intended for instructional use, then proceeds to hold meetings and schedule workshops with the intention of determining what



to do with the new computer/terminal. We suggest that the foregoing approach is backwards and that one ought to begin by identifying instructional problems for which various forms of computing may offer solutions. A five phase model, depicted by Figure 1, provides a systematic approach to the selection of a microcomputer which will meet the present and anticipated instructional needs. (N.B. Many of our analyses inevitably lead to the selection of microcomputers. Obviously, neither all instructional problems nor most of them will lead to computer solution.)

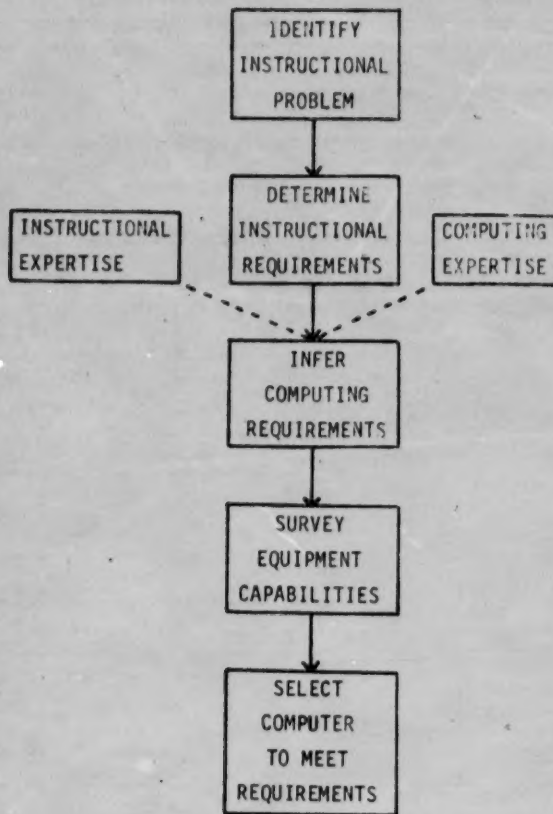


Figure 1 MICROCOMPUTER SELECTION MODEL

#### IDENTIFY INSTRUCTIONAL PROBLEM

Individualized instructional problems require that the teacher often spend time with diverse student ability-groups in the classroom. Some students may work on vocabulary, grammar, math facts, chemical symbols, or spelling. Others need the opportunity to recite, to explain, or to ask questions. We believe that the computer has much to offer the former, freeing the teacher to concentrate on the latter more frequently. It is important that the specific learning activity and a potential computer application be identified early in the selection process. The computer is, after all, a tool which amplifies the teacher's effectiveness. But the teacher, as the respon-

sible professional, must recognize the unique instructional problem for which a computer solution is available. The teacher who realizes that a group of students will profit from interactive drill, on punctuation skills for example, has completed the first step in the microcomputer selection process. Problems associated with the application of rules, or verbal associations, or concept learning could also be mentioned. The teacher's problem identification process could identify many areas where computing could provide help with the learning process. Selecting the appropriate computing tool to solve the problem will yield more satisfaction than fitting the problem to the tools already selected.

#### DETERMINE INSTRUCTIONAL REQUIREMENTS

Having identified an instructional problem, the teacher is best able to select a strategy for teaching the skills required of the student. We have identified nine instructional activities or strategies for which microcomputers might be used. These strategies, although not exhaustive, represent the bulk of applications for which computers have been used in instruction (purely administrative applications have been excluded).

Drill represents a form of interactive computer assisted instruction in which the student responds in a rather quick fashion to brief items or questions under a sort of "flash card" format. The computer provides feedback as to the correctness of the student's answer and may adapt to the individual student by varying the number of items or the frequency of the stimuli as a function of the student's responses.

Tutorial programs provide paragraph-like material, interspersed questions, and response-sensitive branching. These programs are usually characterized by an abundance of textual material.

Problem solving programs typically help the student in the learning of principles or rules by harnessing the power of the computer to eliminate the need for each student to complete complex calculations or engage in the manipulation of multiple logic states. These programs are usually short algorithms for which the student repeatedly supplies data to be manipulated according to the rule or principle being taught.

Programming involves the use of a computer language to provide the proper instructions which control the computer. Perhaps the major single use of computers in instruction is in the teaching of computer programming.

Stimulations model phenomena of an often very complex nature in which random events are introduced to add realism to the interaction. Simulation is a valuable strategy when one needs to compress time (as in an explanation of radioactive decay) or avoid dangerous phenomena or expensive equipment.

Testing by computer permits the educator to exercise options not available with traditional testing methods. Items may be administered contingent upon prior responses to improve test



**314** reliability; alternate tests may be easily constructed for individualized programs; diagnostic tests may be collapsed into short testing sessions; tests may be scored instantaneously.

Computer Managed Instruction (CMI) is characterized by the collection of student data and the transmission of learning prescriptions following objectives-referenced testing.

Data Analysis involves the activities for which computers are most widely known. The use of standardized statistical analysis programs, list processing programs, and similar programs are a crucial topic in many disciplines.

Information Retrieval is a much-used application in the social and physical sciences as well as in the humanities. In this application, the computer is used to select information from a large storage medium according to certain pre-defined rules. Typical applications are literature searches, selective mailing lists, and fact-finding.

Each of these nine activities or teaching strategies has associated characteristics which in part define the instructional mileau. For

example, feedback is a necessary characteristic for drill but not for data analysis; interaction is a typical requirement for tutorials, but not for problem solving programs. Eleven characteristics have been identified which describe the nine strategies described above. These requirements are not exhaustive, but represent what we feel are the major instructionally oriented concerns associated with the activities mentioned.

The eleven characteristics of the instructional "contact" and the nine strategies may be formed into an Instructional Requirements Matrix as shown in Table 1. The intersection of a strategy and a characteristic indicates a requirement which will be translated into a computing requirement. The problem solving strategy may be illustrative: neither feedback nor significant interaction typically take place within a problem solving type program. Graphics may be necessary for some applications, such as where the shapes of mathematical functions are being studied.

The Instructional Requirements Matrix provides the information needed to develop a requirements list. By selecting the strategies which will be employed in the classroom, and collapsing across strategies, one obtains the list of instructional

TABLE 1

INSTRUCTIONAL REQUIREMENTS MATRIX

| Strategy<br>Charac-<br>teristic | Drill | Tutor-<br>ial | Problem<br>Solving | Program-<br>ing | Simula-<br>tion | Testing | CMI | Data<br>Anal. | Info<br>Retr. |
|---------------------------------|-------|---------------|--------------------|-----------------|-----------------|---------|-----|---------------|---------------|
| Feedback                        | X     | X             |                    |                 | X               |         | X   |               | X             |
| Interaction                     | X     | X             |                    |                 | X               | X       | X   | X             | X             |
| Graphics                        |       |               | X                  |                 | X               | X       |     | X             |               |
| Large Data<br>Source            |       |               |                    |                 | X               | X       | X   |               | X             |
| Specific<br>Language            |       |               | X                  | X               |                 |         |     |               |               |
| Multiple<br>Languages           |       |               |                    | X               |                 |         |     |               |               |
| Student Record<br>Keeping       | X     | X             |                    |                 |                 | X       | X   |               |               |
| High Student<br>Terminal Time   |       | X             |                    | X               | X               |         |     |               |               |
| Large Amount<br>of Text         |       | X             |                    | X               | X               |         |     | X             | X             |
| Hard Copy<br>Printed Output     |       |               | X                  | X               | X               | X       | X   | X             | X             |
| Packaged Stat.<br>Programs      |       |               | X                  |                 |                 |         |     | X             | X             |

needs which the computing equipment must satisfy.

In identifying strategies it is important to take a fairly liberal view, considering the relative frequency of use of the various strategies, but also considering the strategies which will be employed in the future. This analysis, by its very nature, is quite subjective, but will provide one with weightings for the various characteristics. These subjective weights are used during a subsequent phase of the selection procedure where specific computing requirements are developed.

#### INFER COMPUTING REQUIREMENTS

Previous phases in the selection process represent basically a needs assessment. The problem was identified and a solution devised which resulted in a set of instructionally-oriented needs as expressed by a collapsed Instructional Requirements Matrix. These instructional needs must be transformed into computer-oriented requirements. This task typically requires the combined talents of two people, one representing instructional and the other computing expertise.

A crucial decision at this stage is whether one ought to be investigating a microcomputer or a larger system. Table 2 shows four levels of computing and recommendations for the kinds of instructional computing which are appropriate for each. The mix of applications, their complexity, and their use-frequency will determine the appropriate system size. From this point forward, we assume that the instructional needs and computing capabilities which have been analyzed have resulted in the decision that a microcomputer is to be selected. The phases completed thus far provide the information needed to focus attention on the microcomputers which are appropriate for our needs.

Table 2

#### COMPUTER CAPABILITIES BY SIZE

- \*VERY SMALL - (KIM, SYM, ELF, Single Board)
  - Programming
- \*SMALL - (PET, TRS-80, APPLE II)
  - Drill
  - Problem Solving
  - Programming
  - Simulation
  - Testing
  - Data Analysis (Small)
  - Information Retrieval (Small)
- \*MEDIUM - (Minicomputers)
  - Tutorial
  - Data Analysis
  - Information Retrieval
  - Other modes
- \*LARGE - (IBM 360, 370, CYBER)
  - All modes
  - Programming
  - Data Analysis
  - Information Retrieval

The following narrative will describe the general capabilities or options of microcomputers that are believed to be essential in the consideration of an instructional computing system. These capabilities will be related to the various instructional activities or strategies. A survey of a range of microcomputers with different features will follow as well as a list of points to keep in mind when making the final selection. 315

#### CAPABILITIES OF MICROCOMPUTERS

At the heart of every microcomputer is the microprocessor, the part of the system that manipulates all information and performs all calculations. Each of the different processors has sufficient power, speed, and precision for most instructional applications, making the brand name less important than some of the other features. However, one processor characteristic that some individuals believe makes a difference is whether it uses an 8 or 16 bit word length. The 16 bit processor might have faster speed, more efficient instruction and data handling, and greater precision. Higher precision might be necessary for some applications such as data analysis or critical simulations; but in most instructional strategies, greater precision is not necessary to aid students in understanding concepts or practicing rules.

The instructional programs and data being processed by the microcomputer are stored in the memory unit. Memory is a vital resource of a microcomputer and the size of the memory is a very important factor in determining the types of instructional computing that can be undertaken. There are two kinds of memory in most microcomputers, read only memory (ROM) and random access memory (RAM). ROM, as the name implies, can be read but not changed, making it non-accessible as user memory. ROM contains code which may not be modified, such as system programs. In some microcomputers, the programming language software is in ROM. However, in most systems, part or all of the programming language software and even some of the system utilities are loaded into RAM along with the instructional programs. Therefore, the number which should be of concern is not the total memory, but the amount of memory available for the instructional materials. For example, on one 16K microcomputer, where K represents approximately one thousand characters, the disk operating system and other system software take up almost 12K, leaving less than 4K of usable RAM. 16K of RAM sounds sufficient but the actual 4K is not enough for the majority of applications. Often the minimum amount of memory listed for a microcomputer is not enough, and additional memory needs to be acquired. Also, the maximum amount of memory that can be supported by the microcomputer should be considered, giving room for expansion. Memory is a very important and relatively expensive resource but advances in the technology are lowering its cost, making larger quantities possible.

Another capability that should be considered is the character set. Many microcomputers have only upper case letters and it is desirable for most instructional applications to have both upper and lower case. Almost all of the other learning materials that a student utilizes makes use of

**316** both. An educator should not settle for less when doing instructional computing. Further, the size of the characters should be noted along with the availability of special characters. The ability for the user to design an alternate character set might be a asset in certain applications and should be kept in mind.

In conjunction with the character set, the number of characters per line and the number of lines per display should be factors weighed in the selection process. The number of lines per display is dependent upon the display unit which is used by the system. A display unit of some kind (hardcopy or video) is required, and if the number of display lines is too small, it can be a limiting factor on the type of instructional application. Also, the number of characters per line can limit what can be output in an efficient and readable manner. An optional size is 64 to 80 characters per line and 24 lines per display for video terminals.

Every instructional system must have some external storage device for keeping the curriculum materials and associated programs. Typically, this storage medium is either a cassette tape or a floppy diskette. The cassette tapes are typical audio tapes and their contents are input into the computer through a regular tape recorder. The floppy diskette is a thin, circular magnetic recording surface enclosed in a square envelope which has an exposed recording area. The diskettes come in two sizes, 8 inch or 5 inch. To confuse matters, the two diskette sizes can also come in double density which holds approximately twice the amount of information of the regular one. The information on a diskette is read into the computer's memory by a disk drive. The disk units have either single or multiple drive capability. If students are utilizing several different programs during a session, a disk capability is recommended over the tape recorder. Tape recorders are inexpensive but are very slow and sometimes unreliable, taking several attempts before successfully loading a program. It is further recommended that at least one microcomputer in a cluster be equipped with a dual disk drive for backing-up the system and creating copies of courseware.

Many microcomputers have graphics capability integrated within the system. In some systems, this feature even includes colors, ranging from four to fifteen different hues. In selecting a microcomputer with graphics, the buyer should consider one with a resolution of 128 by 128 for a minimum. The ability to do graphical representations can be extremely useful in many of the types of instructional computing.

Two other features that might be necessary in some instructional applications are the existence of other programming languages and the ability to utilize the microcomputer as a terminal to another computer. Some microcomputers have available other programming languages such as, FORTRAN, PASCAL, or a CAI language, PILOT. The addition of other programming languages can be necessary in the teaching of computer science. The ability to utilize the microcomputer as a

terminal into a larger computer system can be an asset by permitting the development of programs on the larger system then down loading the resulting code to the memory of the microcomputer. Also, the systems could be used in an hierarchical arrangement where the microcomputer is utilized as a standalone system for most computing but when more resources are required, the larger system could be employed. In an environment where the user has access to other computer systems, it is recommended that the microcomputer be equipped with a communication interface, permitting communication as a terminal.

The last capability of microcomputers to be discussed is the ability to directly connect peripherals, expanding the system's potential. Peripherals that would be extremely beneficial in instructional computing are printers, plotters, graphic input tablets, music boards, speech synthesizers, speech recognition systems, random access audio devices, random access slide projectors, and video disk players. A microcomputer with the flexibility to permit connection of a good selection of peripherals should be viewed favorably. When comparing peripherals, make sure software is available to support them on the particular micro and that their addition will not require a major rewrite of existing courseware. The addition of other media to your instructional computer applications can greatly enhance its effectiveness.

Table 3 shows the capabilities of a microcomputer which are believed to be essential for each of the various instructional strategies. However, a particular application, even though classified within one of the strategies, might require capabilities other than those listed. The educator should compile a set of computing requirements, in conjunction with the Instructional Requirements Matrix, employing Table 3 as an aid.

Several microcomputers having a range of capabilities are described in the survey in Table 4. During the final phase of the selection process, the potential buyer would want to create a similar survey, matching those capabilities and systems that satisfy the computing requirements generated previously.

In completing the selection process, the following questions should be asked of each of the possible microcomputers.

1. Is it flexible and easy to use?
2. Does it have good documentation?
3. What is the manufacturer's reputation?
4. Does courseware exist for it?
5. Can it be expanded?
6. What are the service options?
7. How long is the warranty?

Individuals who own microcomputers are a ready source of valuable information. Additional sources are the various personal computing magazines which periodically evaluate micros, and computer store dealers who offer more than one manufacturer's product. In the area of service, the buyer can choose from purchasing a service contract, sending the system to a factory representative, or



switching boards or chips. If several microcomputers of one kind are purchased, it may be advantageous to purchase several boards or chips and do your own servicing.

Now the selection process is complete and the selected microcomputer should satisfy your stated instructional computing requirements. Having followed the proposed model should give you the confidence in your selection to begin a successful instructional computing experience.

317

Table 3  
MICRO REQUIREMENTS PER INSTRUCTIONAL TYPE

|                                        | DRILL<br>PRACT | TUTOR-<br>IAL | PROBLEM<br>SOLVING | PROGRAM | SIMULA-<br>TION | TESTING | OTHER  |
|----------------------------------------|----------------|---------------|--------------------|---------|-----------------|---------|--------|
| MEMORY<br>(RAM)<br>(USABLE)            | 4-8k           | 16-32k        | 8k                 | 8k      | 8-32k           | 16-32k  | 16-64k |
| LOWER CASE<br>CHARACTERS               |                | Y             |                    |         | Y               | Y       | Y      |
| CASSETTE (C)<br>AND/OR<br>DISK (D)     | C              | D             | C                  | C       | D               | D       | D      |
| GRAPHICS                               |                |               | Y?<br>(CHAR)       |         | Y               |         |        |
| HARDCOPY                               |                | Y             |                    | Y       |                 |         | Y      |
| DECIMAL<br>(NON INTEGER)<br>ARITHMETIC |                |               | Y                  | Y       | Y               |         | Y      |
| OTHER<br>PROGRAMMING<br>LANGUAGES      |                |               |                    | Y       |                 |         | Y      |
| FILES                                  |                | Y             |                    |         | Y               | Y       | Y      |
| COMMUNICATION<br>CAPABILITY            |                | Y             |                    |         | Y               | Y       | Y      |
| NUMBER LINES<br>DISPLAY                | 12             | 24            | 16                 | 24      | 24              | 16      | 24     |



Table 4  
COMPARISON OF MICROCOMPUTER SYSTEMS

|                        | PET             | TRS80               | APPLE              | SORCER-<br>ER     | COMPU-<br>COLOR | ASTRAL          | HEATH<br>H 11  | TERAK        | ORION                     | CROMEM-<br>CO<br>S - 2      | OHIO<br>SCI<br>C2 - 8P   | PDP<br>11/03                |
|------------------------|-----------------|---------------------|--------------------|-------------------|-----------------|-----------------|----------------|--------------|---------------------------|-----------------------------|--------------------------|-----------------------------|
| PROCESSOR              | 6502            | Z80                 | 6502               | Z80               | 8080            | 6800            | LSI 11         | LSI 11       | 8080                      | Z80                         | 6502                     | LSI 11                      |
| MEMORY<br>MIN/MAX      | 8/32            | 4/48                | 4/48               | 8/32              | 8/32            | 8/64            | 4/20           | 28/56        | 16/56                     | 32/512                      | 4/36                     | 32/56                       |
| LOWER CASE             | Y               |                     |                    | Y                 |                 | Y               |                |              | Y                         | Y                           | Y                        | Y                           |
| DISPLAY                | Y               | Y                   | NI                 | NI                | Y               | NI              | NI             | Y            | Y                         | NI                          | NI                       | Y                           |
| CASSETTE<br>OR<br>DISK | C/MD*           | C/MD*               | C*/MD*             | C*                | MD              | DD              | C              | D            | DMD                       | DMD                         | C/D*                     | (DOUBLE)<br>DD              |
| GRAPHICS<br>RESOLUTION |                 | 128 x<br>48         | 280x192<br>40 x 48 | 512 x<br>240      | 128 x<br>128    |                 |                | 320 x<br>240 | 512 x<br>512              | 128 x<br>128                | 256 x<br>512             |                             |
| TEXT<br>(LINES x CHAR) | 25 x 40         | 16 x<br>64/32       | 24 x 40            | 30 x 64           | 16/32 x<br>84   | TD              | TD             | 24 x 80      | 32 x 64                   | TD                          | 32 x 64                  | TD                          |
| COMMUNICATIONS         |                 | Y                   | Y                  | Y                 | Y               |                 |                | Y            | Y                         | Y                           | Y                        | Y                           |
| DOCUMENTATION          |                 | Y                   | Y                  | Y                 | Y               | Y               | Y              | Y            | Y                         | Y                           | Y                        | Y                           |
| MULTIPLE<br>LANGUAGES  |                 | Y                   | Y                  | Y                 |                 | Y               | Y              | Y            |                           | Y                           |                          | Y                           |
| COURSEWARE             | Y               | Y                   | Y                  | MAYBE             | ?               | Y               | ?              | SOME         | SOME                      | Y                           | ?                        | Y                           |
| SPECIAL<br>FEATURES    | CHAR<br>GRAPHIC | COMPLETE<br>SYSTEMS | 4/15<br>COLORS     | MEMORY<br>CARTRID | 8<br>COLORS     | FULL<br>EDUCAT. | 16 BIT.<br>KIT | 16 BIT.      | TOUCH<br>SLIDES<br>PLASMA | S-100<br>QUALITY<br>CONTROL | 6 & 9<br>DIGIT<br>(FAST) | 16 BIT<br>MULTIPLE<br>USERS |
| COST                   | 795             | 698                 | 970                | 895               | 1495            | 4900            | 1295           | 6615         | 10295                     | 3990                        | 825                      | 10500                       |

NI - Not Included  
MD - Mini Disk

DD - Dual Disk  
DMD - Dual Mini Disk

TD - Terminal Dependent  
Y - Yes

\*Extra Cost

## TEACHERS TAKING A BYTE FROM AN APPLE

Earl Keyser  
Instructional Computing Consultant  
Northern Trails Area Education Agency  
Clear Lake, Iowa 50428

This paper describes an implementation of the CAI author language, PILOT. PILOT is an elegant author language, using a minimum of statement types in order to increase the probability of use by teachers. The language has been successfully used with children from the first and second grade level (as users) and the third grade level (as authors) up through graduate students in psychiatry. The specific implementation covered in this paper is on an APPLE II microcomputer.

All was in darkness. On the first day, Von Neuman proposed the computer and there was light. On the second day, Grace Hooper advanced the proposition of higher level languages. On the third day, others advanced the proposition that all languages are NOT created equal. On the fourth day, the concept of "an author language" was conceived, and on the fifth day, Coursewriter was born. On the sixth day, much courseware was developed in Coursewriter and its offspring: TUTOR, IDF, and DECAL. On the seventh day, computer and teacher (who was by now thoroughly exhausted) rested.

The above scenario, although not strictly according to fact, does convey the spirit of the movement from machine language to author language. It also, hopefully, makes the point that author languages can, by their complexity, be as much a hindrance as a help in the writing of instructional materials. Many teachers, particularly those in the humanities, WILL NOT write materials in a language which they feel forces them into a mold they consider inhumane or which, through sure complexity, overwhelms them.

In 1970, Dr. John Starkweather of the University of California at Berkely, saw a clear need to alleviate some of the above pain. His solution was a language which fulfilled the principle requirements of an author language while remaining elegantly simple. This language was called PILOT (Program Inquiry Learning Or Teaching). It was initially implemented on IBM hardware as an interpreter in PL/1 and has since been rewritten for other operating systems in other languages

such as: HP BASIC, DEC BASIC PLUS, FORTRAN II, and PASCAL as well as in a variety of machine dependent languages. Starkweather's original conception has proven extremely durable and able to weather all of these translations without degradation of the original concept. In fact, some materials originally written in Coursewriter at UCSF have been translated into PILOT. Academic authors have, almost without exception, been delighted at the ease of learning and use of PILOT.

In 1973, Dr. Starkweather met with a group of educators from both the public and the private sectors in California and a standard called PILOT 73 was agreed upon. This standard remains with specific instructions on how extensions are to be created. Within the next three years, a nationwide standard will hopefully be created.

In 1976, Dr. Starkweather received a contract from the National Library of Medicine to implement PILOT on a microcomputer. Because of its then current availability and popularity, he chose to work with the INTEL 8080 micro-processor chip. Starkweather implemented a version of PILOT containing all of the commands of PILOT 73 along with appropriate error messages, diagnostics, editing capabilities and I/O patching. Since that time, PILOT has been implemented on MOTOROLA 6800 based systems such as that of Southwest Technical Products Company, on Z-80 based systems such as the Radio Shack TRS-80 and on 6502 based systems. The implementation described in this paper runs on a 6502 based microcomputer, the APPLE II.

**320** The core set of PILOT instructions agreed upon in 1973 contains eight instructions. This set of instructions allows user input, computer output, comparison of input against stored answers, control statements and a minimal computational ability. When PILOT is formally standardized, it will doubtless also contain statements to provide graphics and implement file capabilities.

The present implementation covers the eight basic commands: TYPE, ANSWER, MATCH, JUMP, USE, COMPUTE, END, and REMARK. In addition, full access to the low resolution graphic capabilities of the APPLE II have been provided as well as several routines to create musical tones and add extensions appropriate to the user. A small base of quality instructional material is being developed for or modified for use with APPLE/PILOT. Some of these materials were originally implemented in BASIC, although clumsily, and are more appropriately done in PILOT.

#### Basic Syntax:

PILOT contains eight core commands which allow input/output and answer processing. Every line of a PILOT program consists of a one letter command (sometimes followed by a conditioner), a colon, and an object. An example would be: "T:Good morning, bright eyes."

The eight PILOT statements fall naturally into three groups: input/output, control, and special purpose.

#### Input/Output Statements:

There are three input/output statements: T, A and M.

"T" or Type allows material to be printed at the user terminal. It is precisely identical to PRINT in BASIC or FORTRAN. Numeric and string variables may be embedded within the line and their values will be printed appropriately. String variables are preceded by a dollar sign and numeric variables are preceded by a pound or number sign. For example:

T:Good morning, friend.

Or

T:You got #K correct, \$NAME.

"A" or Answer is the normal mode of accepting input from the user. The input may be a string or numeric variable name by preceding it with the appropriate identifier - \$ or #. This

input may be referenced in a later TYPE statement.

A:

Or

A:\$AARDVARK

"M" or Match is the real power of PILOT. It allows the user's answer to be compared with a correct answer which is stored as the object of the match statement. Match is intelligent and looks for the answer within the user's answer. It may also be, in some versions, a very sophisticated procedure which works for exact matches, inexact matches and partial matches (by using a "don't care" character).

M:Happy, sad, delig, Joy

If one of the answers following the M in the match statement is located within the student's response, a flag is set to true. If no match is found, the flag is set to false. Until another match statement is executed, the author may make statements conditional upon that match by placing a Y for (Yes, a match) after a command or an N (for No match) after the command.

TY:Correct!

This statement will be executed if a match has been found.

#### Control Statements:

The major control statements in PILOT are: J, U and E. These provide the necessary power to move within the program. There is no presently agreed upon method of accessing material outside the program, be it by chaining to another program or the use of a basic subroutine. APPLE/PILOT addresses this problem, with the knowledge that the exact procedures may need to be changed in the future.

"J" or Jump is the direct control statement allowing movement ahead or back within the body of the lesson. Because PILOT has no line numbers like FORTRAN or BASIC, jumps are made to specific places within the lesson designated by a label - an asterisk (\*) followed by a letter or word. Any label may be jumped to, but a jump to a nonexistent label will customarily abort the lesson.

T:Hello, \$NAME.

J:\*Dolphin

T:This line will be invisible to the student.

\*Dolphin T:First, let's introduce ourselves.

T:My name is P.I. Lot.



In the above example, the first line will be typed to the student, the second line will cause an immediate jump to the fourth line which will be typed to the student, and the lesson will continue from that point. Line three will never be seen by a student.

"U" or Use is identical to Jump but will also function as a subroutine. Upon encountering an E or END statement, execution of the lesson will pause and a jump will automatically be made back to the USE statement where execution will continue at the line following the "U" statement. This is precisely analogous to the GOSUB - RETURN in BASIC.

T:How old are you?  
A:\$Age.  
U:\*Dolphin.  
T:I am happy that you are  
\$Age years old.  
E:  
\*Dolphin M:Two, 2, TWO, two  
TY:Amazing, so am I.  
E:

The "E" or End statement is the terminator. If this is the final statement in the program then it is treated precisely as an End statement in other languages. If it is within the program and reached via a "U" statement, then it serves the same purpose as a return in BASIC.

#### Special Purpose Commands:

The last two commands in PILOT are special purpose commands allowing commenting and mathematical calculations.

"R" or Remark allows the user to insert a comment or two in his lesson which is visible only upon listing. It is never seen by the student.

R:You'll never, ever see  
this line in your  
lesson.

"C" or Compute is used for elementary mathematical calculations. Customarily, 26 mathematical variables - A through Z - are available. The values may be modified according to course needs, tested for conditional execution of statements and displayed in T statements by preceding the variable name with a pound sign.

C:A=7  
C:B=A-5  
C:D=A+B

#### Modifiers and Conditional Execution

All statements may be set for conditional execution in two ways. The

first of these is dependent upon the latest match statement.

When a match statement is executed and a match is found, a match flag is set to one. If a match is executed and no match is found, the flag is set to zero. Depending on the result of the LATEST match, statements may be conditionally executed by placing a Y immediately after the command. For an example of this, please look at the example of the U statement where you may see that the response indicating that the computer is also two years old will only be typed if a valid match is found.

Some versions of PILOT on larger computer systems also allow full conditional execution of statements by allowing the command to be followed by a conditioner within parentheses. Typically, this conditioner is of the type (a=5). If the condition inside the parenthesis is true, the statement is executed. If not, the statement is ignored. The present implementation allows the parenthesis to contain only the name of a single mathematical variable. If the variable is nonzero, the statement is executed. Otherwise, the statement is ignored.

T(J):Think spring!

#### Extensions:

The principle allowed extensions to PILOT are one-letter extensions, although none have been agreed upon by any large group of people. Because of the uniqueness of the APPLE, three such extensions have been made.

"S" or Sound has been implemented to allow the creation of musical tones. The syntax of this statement is: S:VAR/VAR. The first numeric variable indicates the pitch and the second variable indicates the duration. This musical tone is then produced at the APPLE speaker.

"G" or Graphics was also an obvious extension to PILOT because of the APPLE's uniqueness. In addition, availability of graphic terminals will become a major addition to CAI. Any CAI language which does not support graphics may follow the dodo and the carrier pigeon.

GH and GV plot horizontal and vertical lines, respectively. GL will (when implemented) plot a line in the high resolution mode of APPLE graphics.

A final nonstandard command implemented because of a need for a return to text mode and a way of clearing the screen is the NO GRAPHICS or NG statement. Its syntax is NG:.



APPLE/PILOT:

These eight basic commands plus the three specifically implemented for use with the APPLE II provide sufficient power to create good instructional material. The teacher may quickly create materials of specific instructional value to his/her students.

It should be noted that this implementation is specifically written in APPLESOFT II, the floating point BASIC available on the APPLE II. Because of the need for certain functions available in APPLESOFT II, this language was chosen over the more quickly executing integer BASIC. It is the belief of the author that great speed is no special virtue in instruction and that a little anticipation may, indeed, enhance the instructional process. The implementation is actually an interpreter, interpreting a lesson and executing it through BASIC statements. Work is now underway on a translator which will allow the author to write, edit, and work in PILOT. Then, when all is in readiness, he/she may convert the lesson into BASIC at which time execution speed becomes a nonexistent problem. In addition, some method of file input/output will be implemented, although this, too, is not a standard in PILOT and, perhaps, is not an appropriate extension of PILOT.

The PILOT Exchange, 22 Clover Lane, Mason City, Iowa 50401, maintains a catalog of instructional materials and the names and addresses of authors and supporters of PILOT. Most materials stored there are intended to serve as examples of what the author may be able to share with others interested in his/her area of expertise. Versions of PILOT for most microcomputers and many larger systems are also cataloged and information is available on their implementation. The present version of PILOT on the APPLE II is commercially available through the Exchange.

Self-Concept:

Fifteen years of research have indicated to us that CAI definitely increases the speed with which materials are learned. Likewise, we can be pretty sure that students retain material as well through the use of CAI as through other modes of instruction. And, of course, we know the computer is a tremendous motivator.

But, one area that remains uninvestigated is the area of self-concept. After a student becomes involved in CAI, does he feel better about himself, about his subject and about school?

Might this be an equally valid argument for CAI if it is true, as my intuition and experience suggest? The present study will examine these questions and tentative results will be reported at the conference.

Summary:

PILOT, a humanistic author language, has been described with a lengthy set of examples showing the basic command syntax and structure of the language. In addition, a specific implementation on an APPLE II microcomputer has been mentioned with appropriate examples. It is the belief of the author that the use of PILOT with microcomputers in schools will provide a new way of using CAI to enhance teaching but also to improve learning. "To boldly go where no man (woman or computer) has gone before" should be the watch phrase of future CAI authors and users. Hopefully, PILOT will help to lead the way.

**DATA - CENTER ADMINISTRATION**

THE CASE OF AN EDUCATIONAL DATA CENTERS  
ENVIRONMENTAL TRANSITION: ITS  
AFFECT ON PEOPLE, PLACES, AND THINGS

Philip A. Mackey  
Delaware Technical and Community College  
Wilmington Campus

Conversion of a computer system to new and supposedly better hardware and software, can be a joy or it may well become a horrible experience. This presentation will point out some of the pitfalls involving, selection of personnel, preliminary and detailed investigations, management involvement, selection of the new hardware, manufactures support, facilities, and implementation. The presentation is based on actual experience and, it is hoped, will aid those who are considering conversion, those who may have just started conversion, and perhaps those who have just recently converted. Reference to a particular computer manufacturer has been purposely avoided in the name of professionalism.

"User endures eight-month switch to DDP (Distributed Data Processing system)"---  
--"the time it took central site batch programmers to grasp the nature of on-line systems". A quote from Computer World, November 27, 1978, Volume XII, No. 48.

It is my intention to reach those who may be thinking about an upgrading of their present computer system or contracting with a servicing institution. I wish to share my experiences both good and bad. Mostly I want to point out some of the pitfalls that may confront you. It is my sincere hope that my experiences will assist you in making the conversion process more efficient and less costly. All users and all providers of educational data processing will eventually come to the conclusion that their system is no longer adequate enough to provide the proper service. The initiation of action toward conversion will be made when the user realizes the inadequacy of his system.

Once there is complete agreement on the inadequacy of the system, the emphasis shifts to the following:

1. Complete and proper planning.
2. Scheduling.
3. Involvement of personnel.
4. Implementation.
5. Testing.
6. Follow-up.

At this point getting started on the right foot is essential. By this I mean, that convincing the administration of the need for a new and better system may well be the first pitfall.

One of the best approaches is the formation of a Computer Users Committee to determine if there really is a problem.

If problems are identified, a study is begun to find solutions and alternate solutions to correcting the problems. The planning aspect begins to take form. Authorization, usually requested by the Data Processing Department, for a feasibility study should be the next move. The administration should be approached by the Computer Users Committee with the facts and figures gathered, (problems and solutions). A decision must be made whether or not further study is necessary. If the administration, at this point, is not convinced\* then there is no need for further discussion. You must live with what you have.

If authorization is granted, the feasibility study should begin. Everyone from students, to department heads, registrar, admissions people, business office manager and any other user of the computer facilities must be involved. No one should be overlooked, if meaningful results are to be achieved during this study. Usually, if this approach is taken, the results are positive. The attitudes of the people involved are positive, and the general over-all atmosphere is positive. This will also promote easier acceptance of a negative result to the feasibility study. Assuming that the results of the study are positive, and administrative approval is obtained, conversion is underway. Many months must be spent now, in meeting with all concerned, conferences, planning, scheduling, and documenting, until an overall conversion plan is developed.

---

\*The most usual justification for "no new system" is administrative costs or budgetary restraints.

**326** Also, contact should be made with the prospective computer manufacturer, communicating the fact that a conversion plan has been formulated by your institution. It is not in the best interests of your particular needs to call in the manufacturer to do the conversion, because systems experts easily influence us by their dynamic appearance and seemingly endless knowledge. Furthermore, we may end up with a system that we know very little about, except that it works. We must also make sure that the system agreed upon will meet not only the present needs for upgrading but also allow for future expansion. A system tailored down to fit costs and/or memory limitations is self-defeating. It is like planning for failure. Let it be known at the onset that you, as the purchaser of computer hardware and software, will maintain control at all times. Overall planning should start at least one year prior to conversion and any commitment to purchase. The final result should be a decision on what would offer the best service. It could mean a new computer or service from another institution, keeping within reasonable cost limits. Remember that over-restrictive cost limits could produce a "lame duck" system.

Education, or perhaps re-education, of personnel is an essential part of the conversion planning. It is, without a doubt, the most difficult phase of conversion to achieve, due to lack of released time and the proper financial backing. Several methods may be employed.

1. Actual off-site classroom type education using the facilities of the manufacturer. It is, however, the most expensive method.

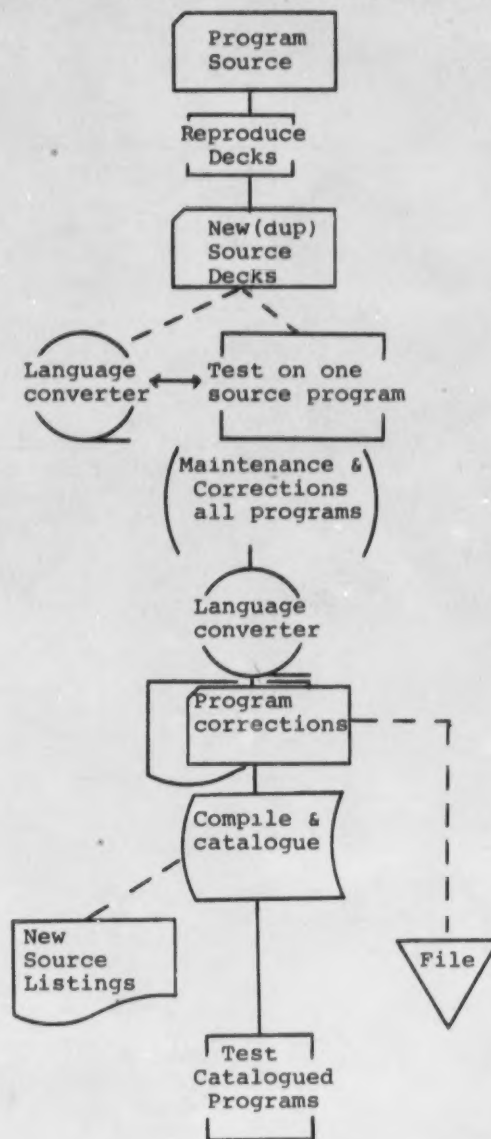
2. In-house education with the use of software materials and manufacturer's advisory personnel. This would be less costly but perhaps not as efficient overall.

This latter method would involve not only key data processing personnel but should also include all users and even administrators. It is absolutely essential that all persons directly or indirectly affected should understand the software and the potential of the new system. This education process must be thorough and on going to be effective.

In the actual conversion phase the use of conversion aids are tools that can be very effective in cutting time and costs. Ask for the most up-dated systems generation software, if it is necessary, since something new and faster has most likely been developed. If programs, of any substantial amount, must be re-written, employ the use of any available language converters. Re-writing programs may be the best method but it is very time consuming and costly.

Language converters are not 100% effective. Depending on the specific program

language, some testing on a small scale must be done prior to program conversion and perhaps necessary maintenance of all programs can be accomplished to increase the efficiency of the language converter.



Make sure that the new equipment adheres to the standard of the existing system. For example, can the data base be converted without any major redesign? Will the standard and preprinted forms need any redevelopment if a new and faster printer is part of the new system? Will the console dialogue of the existing programs adapt to a CRT if a hardcopy now exists? All of these questions and others I may not have mentioned may be pitfalls on the path of conversion. Thorough testing prior to actual implementation, will turn up most, if not all, of this type of



problem.

At the actual point of conversion, all key personnel must be involved, each with their own parcel of the conversion to accomplish. This will also include supportive personnel, even if they only advise. All backups of tape, disk and also card files, should have been completed at this point.

Audit trials should have been established including record counts and any necessary hash totals. The conversion plan must be followed to the letter, taken step by step, one phase at a time. No phase is started until the previous phase checks out. The conversion plan should not be altered unless a major problem occurs. The job will take many long continuous hours of uninterrupted processing and hard work. Personnel involved should be made aware of their responsibilities, they should know what to expect and the responsibility does not end even when the new system is up and running.

Complete testing of all program applications must be done after initial conversion of program, software, and data bases before any live processing can be started. To convert and start live processing without testing, could prove to be a most regretful mistake.

Parallel processing is another method that may be undertaken if the old system remains after the new system (hardware) is up and running. This is also a costly and time consuming method.

Once the conversion plans have been finalized and conversion has been successful there are still many months of follow-up activities to make certain that all was done well.

Then gather together all those hard working people and have a conversion party! I felt a need to relate points gained from my experiences with this subject. Being part of this difficult type of transition has strengthened my belief that both business and educational environments can achieve either success or failure depending on their planning. The result of a well-planned and well-executed computer systems conversion can be most gratifying to all concerned when it is successful.

It can be a humiliating and devastating experience when it is not carefully planned and executed. Most of all, it takes total commitment and team effort to support a successful conversion effort.

## Sample Conversion Outline:

### I. Active Student Data Base -

1. All transactions and processing will proceed until cut-off date.
2. Accumulate counts and totals, at point of processing cut-off.

JUNE 15

### III. Preparation of source programs.

1. Maintenance and corrections.
2. Update duplicate source decks.
3. Run through Language Translator.

JULY 1

### VI. Conversion of all files.

1. Student Master.....7/30
2. Master Schedule.....8/1
3. Mailing Labels.....8/2

### X. Preparation for removal of old system (hardware).

1. Parallel run - all applications.
2. Audit.

### XI. Removal of old system (hardware)

AUGUST 31

## CHOOSING A CLASSROOM MICROCOMPUTER

Ed Herstein  
Community High School  
Ann Arbor, Michigan

Educators are becoming increasingly interested in acquiring microcomputers for their classrooms. Sometimes they have specific applications in mind, but often they will not fully determine the uses to which a microsystem will be put until they have used it extensively in a teaching situation.

How then can educators intelligently select the computers best suited for purposes that cannot entirely be foreseen? This paper will suggest several criteria -- ease of use, variety of input and display devices, expandability, and others -- upon which a well-reasoned decision might be made.

Computers belong in the classroom - that much has become clear to educators who have participated in the "microcomputer revolution" over the last four years. In that short time, small, inexpensive microcomputer systems have grown from un-assembled kits with rows of red lights and toggle switches to plug-in appliances that include elaborate keyboards and sophisticated video displays. Teachers who originally became interested in using micros for digital electronics or advanced programming quickly realized that they possessed machines of virtually limitless potential in a wide variety of disciplines and learning situations.

As one of the educators fortunate enough to have become involved with microcomputers almost from the start, I am frequently approached by colleagues who are interested in acquiring classroom systems of their own. Some have specific uses in mind - to teach programming, to introduce computer literacy, or to conduct laboratory, social studies, or business simulations. But even among these educators, and certainly among the majority with whom I have spoken, there is a strong sense that a microcomputer would be a valuable learning tool even if its precise applications remain to be defined.

I encourage this open-ended approach as both honest and appropriate when dealing with a technology so new and so powerful. Even the most sophisticated examples of instructional computing today barely suggest the promises that will be realized over the next few years. The door

is wide open for any educator with interest and imagination to make a contribution.

By what criteria can an educator select a microcomputer system whose role in the classroom will be fully defined only after it has been put into use? Our experiences at Community High School have begun to suggest some answers. Over the last three years we have worked with Altairs, Apples, IMSAIs, PETs and Scorerers equipped with a variety of terminals, video display boards, mass storage devices, and other peripherals. We have found that the systems that have served us best have shared a number of characteristics that are likely to continue to be important as instructional computing evolves:

THE SYSTEM OFFERS SEVERAL MEANS OF INPUT AND DISPLAY. We have learned that joysticks and other analog devices add an important dimension to a computer's capabilities, and we are eager to explore the merits of light pens, digitizers, and other alternatives to keyboard data entry.

For output, video graphics are essential. It should be possible to add a printer and to interface a video monitor. We are just beginning to experiment with sound and color, and hope soon to discover the educational utility to be found in controlling external devices such as lab instruments and appliances.

THE COMPUTER IS EASY TO OPERATE AND EASY TO PROGRAM. Most microcomputers use the BASIC language, but there are still

important differences. Loading and saving programs is much more difficult on some systems than on others. It is too likely on some microcomputers that a user will inadvertently terminate a program by pressing the wrong key on the keyboard. Screen editing capabilities are sometimes available that make entering or modifying programs much easier. Perhaps most importantly, the languages on some microcomputers have been extended to simplify generating special effects such as color, animation, and sound, while on other systems coding is so complex and documentation so poor that these features are accessible only to highly experienced programmers.

THE COMPUTER IS BEING USED IN A VARIETY OF NEW AND EXCITING WAYS. Some systems seem to attract particularly innovative and public-spirited programmers. Their work in turn is shared and becomes the basis for still more and better software, and the educator's resources grow accordingly. A manufacturer who encourages users' groups and supports contributors' libraries can also do a great deal to promote the dissemination of quality materials.

THE SYSTEM CAN EASILY BE EXPANDED. We have already outgrown the memory we started with on some of our systems and have realized the importance of being able to add more. New peripherals - printers, plotters, communication interfaces, and devices still unforeseen - are constantly entering the market. As their usefulness becomes apparent, our experiences suggest the desirability of being able to integrate such hardware without great difficulty or expense.

THE SYSTEM IS STURDY AND RELIABLE. Some microcomputers overheat or have keyboards that wear out rapidly. Some cannot be moved around without risking malfunction. Sometimes the most desirable features of a system - high-speed cassette interface or timesharing capability - are too unreliable for classroom use. The potential for accidentally damaging a system is often too great, as when a diskette can be ruined if a specific sequence is not followed when it is loaded and removed from a disk drive. We have lived with computing as an elite activity long enough; today's microcomputers should not have to be off-limits to inexperienced users.

THE MICROCOMPUTER IS A REAL PRODUCT AND SUPPORT AND SERVICE IS AVAILABLE LOCALLY. Many of the most attractive systems exist only as prototypes for months after they have been announced. Systems ordered by mail will usually have to be serviced the same way. Any user of a system will occasionally have questions

that only someone associated with a manufacturer can answer. The system that is sold and serviced locally and is backed by a manufacturer who treats both dealers and owners responsibly will not wind up gathering dust the first time a user has a problem he or she cannot solve without help. 329

Unless they are relevant to the above considerations, I have little concern for such technical issues as the type of processor, memory chips, or bus structure that a system may use. Price is certainly a major consideration, but the sales price is often deceptive. The cost of a system is not just the selling price but also the expense of adding peripherals, acquiring software, and providing maintenance. Each of these must be investigated carefully before an accurate price comparison can be made.

Finally, I would encourage anyone interested in purchasing a microcomputer for education to realize that selecting and learning to use a classroom system are in themselves educational activities. I believe microcomputers are going to bring great change to the process of education, and that only by learning through our own experiences with them will we be prepared to deal wisely with the future that lies just ahead.



# THE DEVELOPMENT OF A MANAGEMENT INFORMATION SYSTEM: A CASE STUDY

Edwin D. Bell  
North Carolina A & T State University

This paper will describe the development of a computer assisted management information system at a historically black state institution, which has become part of a consolidated university system in the last ten years. The activity was started as part of the Institution's Advanced Institutional Development Program and was not really well understood throughout the University. The activity was based on the traditional model of MIS development, i.e. a system analyst produces designs and applications to meet the needs that the users have identified. For a variety of reasons this particular strategy was not successful. The failure was analyzed and a new strategy was produced based on some of the operating assumptions of Organizational Development. The new strategy is working well and the implementation of the MIS is making good progress.

"Anyone who has common sense will remember that the bewilderments of the eyes are of two kinds, and arise from two causes, either from coming out of the light or from going into the light, which is true of the mind's eye, quite as much as the bodily eye; and he who remembers this when he sees anyone whose vision is perplexed and weak will not be too ready to laugh." (Plato in Jowett, 1978, p. 258)

The particular project of planned change that will be discussed in this paper is the development and implementation of a computer assisted management information system, MIS. The proposed MIS would facilitate the routine operation and reporting of all administrative offices and provide timely and accurate information to administrators that they would use in the planning, management, and evaluation of the Institution.

The MIS project is really only part of a more comprehensive change effort to renovate the administrative tools, procedures, and policies of the entire Institution. This comprehensive change effort was labeled Planning, Management, and Evaluation, PME, by the United States Office of Education.

Surprisingly both PME and MIS were to be planned, implemented, and evaluated in only five years. Given the gift of hindsight, it is easy to see that the projected time schedule was shockingly naive, but let us consider some of the other factors which created and constrained this particular MIS project.

Both internal and external factors have had a major impact on the MIS project, but let

us start with some of the relevant internal factors. The project was located in a historically black state supported school in the southeastern United States. The Institution has a long, proud tradition and had within the past ten years become part of a comprehensive state university system. Most of the senior administrators at the Institution have been there a long time and most of the senior administrators of the state's comprehensive system have also been there a long time. The two groups of administrators have a long history with one another and it is probably safe to assume that they have fairly well established attitudes and perceptions of one another. In fact, the General Administration of The University System had gone on record as saying that any current deficiencies in this particular historically black institution were not a result of unfair distribution of resources, but inferior administrators, faculty, and students at this particular Institution.

Although there is no direct evidence, it is reasonable to assume that administration of this particular Institution might feel that The University System was, if not hostile, certainly is neutral to the growth and development of historically black institutions. There are some other assumptions held by the administrators of this Institution that are particularly relevant to the MIS project. First, social and political factors not organizational effectiveness are the main determinants of their institutional growth or decline; and second, they needed to look elsewhere for necessary resources. This last assumption leads us to some of the external factors that affect the project.

One of the other sources that this particular Institution has developed is the Title III Program in the United States Office of Education, USOE. Title III had administratively created a



program called Advanced Institutional Development, AID. (AID is now being phased out.) This program offered multiyear funding for eligible institutions and offered the potential of large, secure grants. It also had several new requirements and suggestions. One requirement was that a grantee spend a certain percentage of the grant on Planning, Management, and Evaluation, PME, which dealt with the modernization of the management of the institution; one suggestion was that the grantee update its management information system, MIS, and consider making it computer assisted. Consequently PME and MIS were incorporated into the Institution's successful proposal, not as a response to perceived internal need but as a response to perceived external requirements and opportunities, and MIS as a planned change intervention was initiated in 1975.

### The Classical Opening

The MIS intervention was planned and conceptualized along traditional lines. A computer assisted system was planned that would rely on Computer Center personnel writing programs or applications that would meet user needs for information. The implementation of the intervention was also traditional and widely supported in MIS literature.

"The first step, then, in implementing a successful computerized data processing program is to gain full management commitment. To do this, the extent to which present operating difficulties are due to lack of good information must be brought directly and forcefully to top management's attention." (Alexander, 1974, p. 112)

This first step implies an explicit frontal assault on the current procedures of the Institution and an implicit assault of the self-esteem of the people who developed, maintained, and used those procedures. Goffman (1963) predicts that people will leave or avoid interactions that cause them to lose "face" or self-esteem. However, for two years the Institution attempted to complete that important first step with limited success. In 1977 MIS was reorganized and reassessed. It was argued that the Institution should be viewed as an open system and that consequently the same end state could be reached from various starting points by diverse routes.

Viewing the Institution as an open socio-technical system implied two concepts:

the institution is made up of the joint operation of two independent systems, a technical one and a social one, and

the institution continually interacts with an external environment which influences and is

influenced by the organization. (Cummings, 1977)

Any proposed intervention would have to consider both concepts. As we continued our diagnosis of the Institution in search of an appropriate intervention strategy, perceived locus of control was identified as a key issue (Korman, 1977). Many individuals believed that what they did had little or no effect on their overall achievement or success. Another key issue was the unwillingness of key administrators to invest too much commitment in something that was merely a technique to acquire necessary resources that could not be acquired through the state.

### The Revised Intervention

After the assessment of the reasons for the slow progress of the MIS intervention was completed. A new intervention was proposed. The new MIS would be a computer assisted, data-based (as opposed to an application-based) MIS. The data base and relevant applications would be implemented through user modules and developed through a bottom-up technique (as opposed to the traditional top-down technique) (Eno & Phoenix, 1977). The intervention and its implementation strategy were focused on four issues of the sociotechnical system: power and authority; morale and cohesion; standards and norms; and goals and objectives of the Advanced Institutional Development grant (Blake & Mouton, 1976).

The intervention strategy had to increase the political effectiveness of the senior administrators without asking them to invest too much prestige in an unproven project. Therefore, the intervention started out with pilot projects that would grow if they proved successful and some of the projects focused on issues that would be useful to the administration of the Institution in their political negotiations with the General Administration of The University System. The intervention had to improve the morale of administrators, not tear down their self-esteem; so a program of job enrichment was initiated both horizontally, a greater variety of tasks with no increase in level of decision making, and vertically, a greater role in decision making (Korman, 1977). The mechanism of job enrichment was a network of advisory and internal consulting teams which received specialized in-service training. Members of the network were also sent off-campus to relevant seminars whenever possible. The norms of the organization were not challenged initially in the selection of any of the committees. The project only started to deviate from established norms when it had some credibility to trade upon. The fact that the Institution had entered into a contract with the USOE in which it had described goals and measurable objectives was the bedrock on which senior administrators and policy groups based their support of the initial inconvenience which the intervention created.

332 The strategy has worked well and the MIS project has received excellent evaluations from its external evaluators, but there are still some unresolved issues.

#### Current Assessment

Morale and group cohesion are still a problem when translated into a sense of personal efficacy and loss of face. A large number of people still feel that external political forces determine what happens to the Institution and their individual effectiveness and efficiency are irrelevant. Also there is still a minority of people who perceive any planned change as criticism of the status quo and a direct threat to their self-esteem.

Another unresolved problem is the sponsorship of MIS, i.e. who is really supporting the activity. All financial support has come from the federal government and the General Administration of The University System has rejected budget requests for state support of the activity. Will The University System be willing to support constructive planned change within one of its institutions?

Acknowledging the fact that these problems will impede the progress, the Institution is confident that it will produce a comprehensive computer assisted, data-based management system that will allow the organization to respond to internal needs and external pressures.

#### References

- Alexander, M.J.. Information Systems: Theory and Applications. Chicago: Science Research Associates, 1974.
- Bell, E. D. Administrative Planning: Science or Art. Paper presented at the meeting of The Society of College and University Planning, Ft. Lauderdale, June, 1977.
- Blake, R. R. & Mouton, J. S. Consultation. Reading Massachusetts: Addison-Wesley, 1976.
- Cummings, T. G. Sociotechnical Systems: An Intervention Strategy. In W. W. Burke (Ed.), Current Issues and Strategies in Organizational Development. New York: Human Science Press, 1977.
- Eno, E. J. & Phoenix, G. J. MIS Development: A Participative Process. Paper presented at the meeting of The Society of College and University Planning, Ft. Lauderdale, June, 1977.
- Goffman, E. Behavior in Public Places. New York: The Free Press, 1963.
- Jowett, B. Plato's The Republic. New York: Vintage Books, 1978.
- Korman, A. K. Organizational Behavior. Englewood Cliffs, New Jersey: Prentice-Hall, 1977.
- Warwick, D. B. Moral Dilemmas in Organizational Development. In G. Bermant, H. C. Kelman, & D. P. Warwick (Eds.), The Ethics of Social Intervention. New York: John Wiley & Sons, 1978.

# THE DIALECTIC OF ARTIST AND TECHNOLOGIST IN INSTRUCTIONAL SYSTEMS DEVELOPMENT

G.P. Kearsley & A.F. O'Neal  
Courseware, Inc.

There is a trend today throughout the educational system toward the use of systematic principles and methods to design, develop, and deliver instruction. This movement is called Instructional Systems Development (ISD) and is an attempt to develop a scientific or engineering approach to instructional activities. It is assumed that, once ISD principles have been sufficiently well specified, most instructional activities can be done by technicians or by automated (i.e., computer-based) instructional systems. Little attention seems to have been given to the limits to which a technology of instruction can develop. Almost all instructional activities depend, sooner or later, upon unique and creative processes. In fact, it may be these artistic contributions which determine whether instructional materials or presentations are effective or not. In other words, the presence or absence of craftsmanship may be the most important quality of instructional development. This paper presents a dialectical approach to the issue of the importance of intuitive processes and analytic procedures in ISD. The outcome of this discussion is a clarification of the role of artistic creation in ISD and a resolution of two opposing viewpoints. It is argued that a renaissance man is needed in modern education--but a different type of renaissance than has been true in the past.

## Introduction

There has always been a need for the "Renaissance Man" in education. Teachers and administrators alike have always had to carry out many simultaneous roles. However, in the past, the essence of renaissance ability has been on possessing the skills to actually do a diverse range of things. Today, and in the future educational system, renaissance will refer to the ability to invent or use tools such as computers to carry out the diverse range of activities associated with the instructional process.

The history of instruction reveals three different periods or stages of development which recapitulate the history of science: the artistic approach, the empirical approach, and the systematic approach. In the artistic approach, the design, delivery, and evaluation depends upon intuition and personal experience. To the extent that one person's intuition or experience was better than another's, better instruction ensued. In the empirical approach, intuition is used to formulate hypotheses which are then used to design experiments and collect data. Instructional methods or techniques which are supported by the data tend to be accepted over those which do not. In the systematic approach, analytic procedures or methods are used to design, develop,

or evaluate instruction. These procedures are derived from a combination of refined experience, accumulated research findings, and learning theory.

The systematic approach is called Instructional Systems Development (ISD) or just simply instructional development. ISD has emerged in response to the pressing need for a more effective, efficient, and reliable educational system. It involves things such as task analysis, behavioral objectives, criterion reference testing, individualized instruction, formative evaluation, and media selection. While ISD was originally developed in the context of military training, its use has spread to all domains of instruction and is probably the dominant instructional methodology existing today.

A major theme of ISD is that instructional activities (design/development/evaluation) can be carried out by technicians or automated via computers once the appropriate procedures are available. For example, a media selection algorithm can be applied (by a technician or computer) to decide the appropriate delivery medium for a particular instructional sequence. Or an individualized instructional program can be carried out, not by the socratic interaction between tutor and student, but by a course manager and a computer-managed instruction system. Similarly, the test



**334** items can be constructed not by an experienced teacher but by an item generation rule which produces a question which makes the behavior, conditions, and criterion specified in an objective.

Despite the success of ISD in developing a technology of instruction, a major issue has never been addressed by the ISD field. This is the issue of the importance of artistic intuition or craftsmanship in the ISD process. Throughout the design, development and delivery of instruction, there are many occasions where creative effort is required in order to produce effective instruction. For example, clever visuals, novel layouts, humor, imaginative practice exercises, inspiring prose, etc. all may contribute more to the quality of instructional materials than the use of ISD procedures such as task analysis, objectives, media selection, and so on. Mechanical following of ISD procedures does not lead to this creative effort and, therefore, necessarily produce effective instruction.

This article deals with the dialectic between artistic processes and technological procedures in ISD. It follows a classic dialectic approach with a thesis, antithesis, and synthesis. The paper attempts to avoid "strawman" arguments and focuses on the successful resolution of legitimate problems. In doing so, it is hoped that a more robust framework for ISD can be developed.

Thesis: Artistic processes are critical to the development of effective instruction.

1. Every instructional problem is unique and only an idiosyncratic approach at all levels can fully deal with this uniqueness. If this were not the case, then the same processes and procedures could be used for every instructional project with reliable outcomes.

2. The mechanical or automatic application of ISD procedures to the design and development of instructional materials by technicians or computers leads to very "mechanical" materials. These materials may be instructionally adequate but dull, boring or lifeless. The lack of intrinsic appeal of these materials to the learner must be made up for by management or incentive systems. This is why instructional management is so important in ISD.

3. The use of standardized ISD procedures could make the instructional development process tantamount to an assembly line. The quality of the work is de-emphasized due to the attention given to the number of units assembled.

4. The use of prespecified procedures tends to obscure the needs of a particular instructional effort. A technologist or computer concentrates upon applying the known procedures rather than inventing new procedures or techniques which might be needed. ISD can be a solution in search of a problem.

5. The rationale for using ISD procedures over purely artistic, idiosyncratic methods is that they result in more effective or efficient instruction. It is suggested that there are relatively few cases where ISD procedures have resulted in either and there are as many cases where artistic efforts have been as successful.

6. ISD procedures apply only at a very global level and really only serve as a framework for making artistic judgements. Thus, the entire idea of a scientific approach to instruction is fraudulent--in the final analysis, all specific decisions are based upon personal intuition.

Antithesis: Analytic procedures are critical to the development of effective instruction.

1. While every instructional problem is indeed unique, each instructional situation also has much in common with others. The problem of training a pilot to fly a 707 is similar in many ways to teaching a pilot to fly a 747 or any other type of commercial jet. To the extent that these common factors can be identified and tied to appropriate research and experience, the benefits of systematic procedures can be achieved. There is no need to reinvent the wheel for every instructional problem.

2. The application of ISD procedures leads to standardized, uniform instructional materials which are instructionally effective and, therefore, adequate. While artistic creation may lead to some truly outstanding materials, it also leads to many elaborate or intricate programs which are instructionally useless. ISD procedures, therefore, result in reliable instruction. Instructional management is a further step in producing reliable instruction since it ensures that all scheduling and testing activities are properly organized and work.

3. The use of standardized procedures makes the process of designing and developing instruction more like bridge building or surgery. A set of methods and techniques exist which are followed to ensure that the desired outcome (i.e., good instruction) occurs. Idiosyncratic and whimsical deviation from these procedures reduces the chance of producing successful instruction.



4. The use of prespecified procedures allows the instructional developer to focus on the important aspects of the ISD process and ensure that these are done correctly. This means that the critical steps in ISD are always attended to and not left to chance or memory. The use of such procedures also makes it easy to teach someone else to develop instruction -- a basic flaw of the artistic method.

5. The really critical advantage of the systematic methods of ISD over the artistic approach is that successes or failures can be related to specific instructional methods or procedures which are documented. Thus, ISD has the potential to improve from its failures whereas the artistic approach does not.

6. ISD has led to the development of principles for what is known about instructional development. At the present time, much of this knowledge is relatively global. As more research is conducted, ISD procedures will extend to a finer degree of detail and specificity. There will always be decisions which must be based upon intuition, but ISD procedures can reduce these to a minimal number. This will reduce the scope of error and, hence, the management of their consequences in the instructional development process.

Synthesis: Incorporating artistic intuition into ISD.

1. ISD procedures incorporate lessons learned from past instructional efforts and provide a set of tools which can be used to develop effective instruction. The intuitive judgement of the practitioner is required to apply these tools to a particular instructional problem. In other words, artistic creativity is manifested in the selection and application of ISD procedures.

2. ISD practitioners must be sensitive to the limitations of the procedures they use. Mechanical and blind application of ISD procedures is probably equally as bad as artistic stabs in the dark in terms of producing effective instruction. Artistic solutions should be employed in any specific situation where standard ISD procedures have been tried and clearly are inadequate. Artistic processes which make positive contributions to instructional quality should be studied and developed into procedures. For this to happen, artistic processes must be documented.

3. Artistic talent is a very valuable and essential aspect of ISD. However, it is a waste of resources to have this precious artistic talent applied to quite straight forward components of the instructional design, development, and delivery process. For example, a good teacher should spend

time actually tutoring students not scoring tests or scheduling classes. Technicians or computer systems can take care of these mundane chores and, therefore, result in greater artistic productivity. ISD actually allows greater utilization of human potential because it allows artists or technicians to work on instructional activities which are most appropriate to their level of ability, experience, or training. It can be argued that great artists are too valuable a resource to work at all and should spend their time turning their knowledge or methods into procedures which can be followed by others.

4. The days when the majority of instruction can be hand-crafted are as far behind us as the days when clothing was individually tailored or people churned their own butter. The magnitude of today's educational system forces us into a technology of instruction. We can rarely afford the price of instructional craftsmanship. However, we can convert the principles and techniques of craftsmanship into ISD procedures which can be used on a wide scale to design, develop, and implement effective instruction. To the extent we are successful at doing this, we will achieve a high quality educational system.

5. ISD requires a renaissance man but one whose scope and invention encompasses artistic processes and analytic procedures. Thus, modern education requires a broad command and use of the tools which are available to design, develop, and deliver instruction rather than the personal abilities to actually do every step or activity in the instructional process.

## DOCUMENTATION STANDARDS AND THEIR APPLICATION TO COMPUTER SYSTEMS

Frank B. Thomas  
The University of Akron

Good documentation standards have always been a problem to the data processing manager. How much documentation, and what documentation are questions most managers must tackle.

This paper will focus our attention on standard naming conventions to tie a system together, whether they are data files, programs or manual tasks. The benefits of managing a computer center that requires accountability reporting of its resources will be tied to this standard. Although the idea of a standard naming convention is simple, concise and easily applied, I find few data centers that apply such a standard.

Good documentation has always been a problem to data processing centers.

How much documentation?

What kind of documentation?

These are the major questions which tend to plague the managers of data centers, and I find many times that each systems analyst will have his or her own version of what good documentation should contain and what the standards should be. How do standards play a role in the documentation and why standards? Should there be any standards at all? Again, these questions are difficult to answer, yet I find standards in naming conventions, program coding, and systems design are a necessary evil if the credibility of the computer center is to remain high in the minds of the data processing users.

The function of all documentation is to communicate to the appropriate section within the computer center, as well as the external user, the correct procedures to be followed to accomplish the task in as few words and symbols as possible.

This article will attempt to deal with these questions and present illustrations as to how standards apply to documentation.

### System Standards

One function of system standards is to attempt to tie the system together with one standard nomenclature.

Many physical items exist within a system such as magnetic tapes, forms, card decks, programs, run books, etc. If we could assign a standard naming convention that would enable each item to be identified clearly, then one benefit to be reaped by such a standard is the ability to pick up any one of these items and clearly identify the system to which it belongs. For example, any report could be readily identified to tie it back to a particular system and set of computer programs.

To accomplish such a task, a standard nomenclature must be utilized for all items that must be labeled and accounted for within a system. The nomenclature that I found most useful not only tied the "items" of the system together, but also satisfied the following definitions which describe most operational batch systems.

1. A system is implemented in time-dependent jobs called job runs. (Time dependent simply means certain jobs must be run before other jobs can run. For example, we must update the master file before we run the weekly reports, etc.)
2. A job consists of a set of steps to be performed.

3. A step is either automated (computerized) or it may be a manual task that has to be performed.
4. A computer program is an automated step or task.

With these fundamental definitions in mind, we can then define a job reference or (job related) number whose sole purpose is to label all jobs, steps, programs, tape files, disk files, reports, etc., that go into the design and implementation of a system. This job-reference number will also serve as an identifier for internal costing and reporting of computer center resources.

The format of this job-reference number is eight positions with three subgroups. The first three positions identify a system, subsystem, or category that needs unique identification within the data center. The fourth position identifies the job, step, program, report, or file within the system. The last four positions are sequence identifiers with insertion capabilities. The eight positions are as follows:

#### AANBYYYY

identifies the systems or subsystems

Example: PR1 - Payroll (the first Payroll)  
PR2 - Payroll (the revised Payroll system)

#### AANBYYYY

is the object identifier code

Identifier Codes:

- C - Card file number
- D - Disk file number
- F - Form number
- J - Job number
- M - Memo number
- P - Program number
- R - Report number
- S - System or step number
- T - Tape file number
- U - Utility program
- X - Subprogram number

Examples:

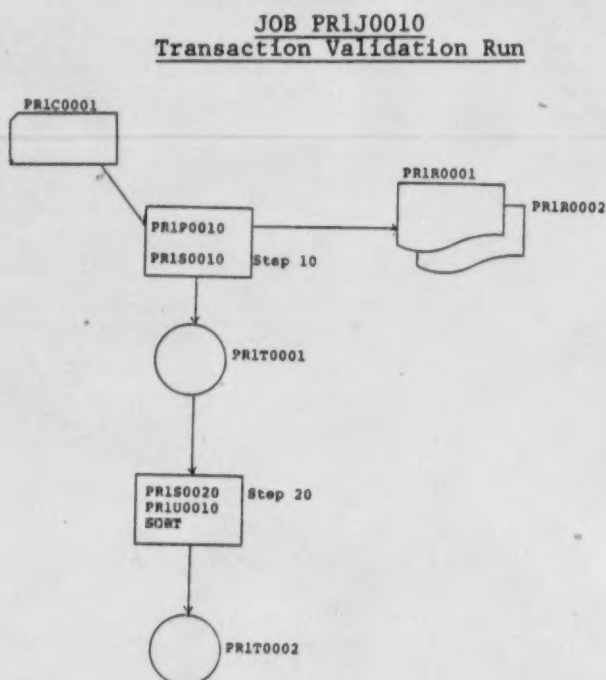
PR1S0000 - Payroll System

PR1P0010 - Program number 10

PR1F0001 - first form design for payroll

PR1R0001 - First report number in the payroll system

With the eight digit job-reference number, we can identify a job run. For example:



This particular job run might process a deck of transactions (PR1C0001) and produce two reports (PR1R0010, 20), an unsorted valid transaction file (PR1T0001), and a utility sort (PR1U0010) that sorts the raw transaction file. Please note the entire job is labeled PR1J0010 and would be documented in both the operator and control section run book. Each program is labeled with a step number to identify application programs that are run as well as utility programs. Also, operators and control personnel identify each task by a step number whether it be a computer program or a manual task. Step numbers are unique and assigned sequentially. Program numbers are not unique and may be utilized in many different job runs.



Examining the job-run flowchart a little further, you will notice the report numbers PR1R0010, 20, etc. This will require the systems analyst to complete a Report Log to identify each report by title and list the distribution. Secondly each file must be identified and a file log maintained. (See the attached sample logs).

This leads us to examine our system as it is designed into time dependent jobs. Our standards have helped us to identify the parts as we design the system. The report numbers, file numbers, program numbers are all recorded on logs to give our systems and programming people a complete list of all the reports, programs, and files that support the entire system. A fourth log of form numbers whether they be input or output pin-fed forms must be captured to insure the system is tied together.

A second benefit that cannot be overlooked is the ability to check the logs for what files, forms or reports are related to what computer programs. How many of us have had to modify our system and forgot one file was utilized in a separate job run that we had completely forgotten?

If these documentation standards are followed, you will have a complete picture of each system within your computer center.

#### What Kind of Documentation and How Much

In this section the documentation will be tied together within the various sections of your computer center. When a set of standards as described above are used, the systems documentation can be broken down to satisfy all the sections within the computer center.

Systems documentation encompasses the three main areas of support, the Systems and Programming Section, Operational Section, and the User's Department. The Operational Section consists of three areas:

Data Entry  
Scheduling or Control  
Computer Operations

As illustrated on the following pages, each section's documentation is based on the needs of that section. Some documentation is duplicated to satisfy that need. For example, report logs are required in the Systems and Programming Documentation, in the Control Section of the computer center for distribution of reports, and finally in the user documentation to let the external user know what reports

are available and when they will be run.

The following documentation is a suggested breakdown by section and can be adjusted to meet the needs of your organization.

#### SYSTEM DOCUMENTATION

##### A. Systems and Programming Documentation

1. Feasibility Study
2. Specification Document
  - a. Procedures/Maintenance/Run Request
  - b. Controls
    1. manual
    2. computer
  - c. Record specification forms
3. System Documentation
  - a. System flowchart
  - b. Job run flowcharts
  - c. Input forms and formats
  - d. Output forms and formats
  - e. Output reports
  - f. Master file layouts and data logs
  - g. Report log
  - h. File log
  - i. Program logs
4. Programming Documentation
  - a. Program flowcharts
  - b. Program specifications
  - c. Program listings
  - d. Sample inputs and outputs
  - e. Work file layouts

##### B. Operational Documentation

1. Library
  - a. Assigned space for disk
  - b. Rotational procedures for tape
2. Control
  - a. Run request/scheduling procedures
  - b. Deck setup or JCL listing by program and by job
  - c. Report logs and distribution
  - d. Logging and balancing procedures
  - e. Job run flowcharts
3. Key punch
  - a. Sample form and card layouts
  - b. Key punching and verifying instructions



4. Operators
  - a. Job run flowcharts
  - b. Operating instructions for each step
  - c. Error and recovery procedures for each step
  - d. JCL listing by program

#### C. User's Documentation

1. Job name and number
2. Description of each job
3. Run request procedures
4. Schedule run log
5. Report logs
6. Description of each report with sample outputs

#### Benefits of Standards and System Design

Not only does the system prefix identify the items within a system, it goes hand in hand with job accounting so that your job accounting reports by job run or program steps are now formalized and the user community has more confidence in the computer center's reporting. No longer will you see such programs or steps named "HARRY'S - WONDER" or "MARGE'S - BEST."

A second benefit utilizing this standard is that manual procedures within your center can also be documented in the same manner. Each manual set of steps can be related to a unique job number. Similar forms utilized for standard practices can be assigned a system prefix. For example, our Center utilizes several types of standard forms in the documentation of each system. We assign such forms as Program Logs, Report Logs, Operation Trouble Report Logs with a general form number that has a system prefix PPl (Practices and Procedures).

The benefits from standard naming conventions for all systems within a computer center is immeasurable. We now have a handle on every form we utilize, every program, tape file, etc.

By tying this standard naming convention into our job accounting system we have a better management tool for the control of disk and tape files, as well as assistance for system project control. The development cost of each system is captured, and this cost then separated from production costs. We can also utilize these standards in our tape library and disk library systems as we relate the usage and space

reporting back to the system that requires these files.

In summary, I have attempted to relate one specific naming convention to tie the parts of a system together. The benefits of these standards reach out to other computer-related subsystems that control the everyday management of our center and provide us the good benefits of documentation standards as well as the side benefit of a high credibility factor by the center's external users.

| THE UNIVERSITY OF AKRON COMPUTER CENTER              |          |                         |                 |
|------------------------------------------------------|----------|-------------------------|-----------------|
| REPORT LOG                                           |          |                         |                 |
| REPORT NO. & NAME                                    | STEP NO. | FORM NO. & INSTRUCTIONS | DISTRIBUTION    |
| PR1R0010<br>Validation Error Listing                 | PR1S0010 | Single                  | Control Section |
| PR1R0020<br>Control Totals for Validation Error List | PR1S0010 | Single                  | Control Section |

PP1F0008  
Rev. 11-35-73

## THE UNIVERSITY OF AKRON COMPUTER CENTER

## PROGRAM LOG

| Program ID | Program Name               | Program<br>51 Language | CROSS REFERENCE      |                    | Notes             |
|------------|----------------------------|------------------------|----------------------|--------------------|-------------------|
|            |                            |                        | 61 To Files          | To<br>710per-Steps |                   |
| PR1P0010   | Payroll Validation Program | COBOL                  | PR1C0001<br>PR1T0001 | PR1S0010           | Valid Trans File  |
| PR1V0010   | Sort Utility               |                        | PR1T0001<br>PR1T0002 | PR1S0020           | Sorter Trans File |

PP1F0006 4/73

## FILE LOGS

| FILE NUMBER | FILE NAME                | DESCRIPTION                                             |
|-------------|--------------------------|---------------------------------------------------------|
| PR1C0001    | Raw Transactions         | Time Card Transactions into Payroll Validation.         |
| PR1T0001    | Validated Transaction    | Only Valid Transactions are placed on this file.        |
| PR1T0002    | Sorted Valid Transaction | Only sorted valid transactions are on this file by SSN. |

# ESTABLISHING A CAI CENTER AND DEVELOPING SOFTWARE: PROBLEMS, CONCERNS, AND RECOMMENDATIONS

Thomas S. Jordan, Ph.D.  
Assistant Director, CEL  
Cleveland State University

This paper explores the historical development of a CAI center at an urban medium size university. The problems in the development of a CAI center (hardware and software) from its inception to operation by the efforts of faculty and an Instructional Development Center are examined. Specific recommendations serve as guidelines for faculty and administrators of postsecondary institutions involved in establishing or improving a CAI center. Recommendations concerning software center around the theory and principles of CAI instructional design. Recommendations concerning hardware are primarily directed toward internal modifications of an existing computer system and CAI facility specifications.

Many lessons can be learned from tracing the significant historical events that have led to the present state of affairs. It is the purpose of this paper to explore the roots of an existing Computer Assisted Instruction Center (CAI Center) to distill from the past lessons that might aid faculty and administrators in initiating or developing their own CAI center. Those factors that inhibited and facilitated the development of the CAI Center at Cleveland State University (a medium sized urban university) are considered. However, the recommendations which follow such consideration are not intended to be broad sweeping generalizations, rather the applicability of the recommendations to other institutions should be scrutinized in light of institutional characteristics.

## The CAI Center Now

After 2 1/2 years since the idea of a CAI Center at Cleveland State was born, approximately 350 students from 15 different courses have received CAI. During the last quarter alone 677 hours of CAI was made available to eight faculty members. IBM's Coursewriter III supported by an IBM 360 computer and four Courier CRT terminals hard wired to the main frame provide the system's hardware. The CRT's are located in the Media Lab, which is a division of Instructional Support Services (ISS) at Cleveland State. Other divisions within ISS include the Center for Effective Learning (CEL) and Instructional Media Services (IMS). All CAI efforts by faculty are coordinated by the Assistant Director of CEL. CEL is the on campus instructional development agency and works closely

with the Computer Center in supporting the CAI Center. The Assistant Director of CEL provides instructional design expertise for faculty wishing to utilize CAI and a part time CIS student performs the necessary computer programming. Personnel in the Media Lab are responsible for scheduling and registering students in addition to posting any procedural instructions for a CAI course.

During the Fall Quarter, 1978, a survey of students' attitudes toward their CAI experience was conducted by Thomas S. Jordan, Assistant Director of CEL. A pre-attitude assessment was made before students experienced CAI and a post assessment was administered upon the student's completion of the CAI course. Three CAI courses were evaluated. For all of the courses a positive significant shift in attitude ( $p < .05$ ) was determined. The results are shown in Table 1.

Table 1  
t-Test Results for Students' Change in Attitude toward CAI

| Course  | df | t     | p (1-tail) |
|---------|----|-------|------------|
| MUSCON  | 8  | +3.18 | .01        |
| LOGIC 1 | 26 | +3.83 | .005       |
| ENGCOM  | 8  | +3.64 | .005       |

A survey of faculty using CAI showed that they rated CAI services above average. As of Fall Quarter, 1978, eight faculty have been involved in the development of six different CAI courses. In addition, approximately 15 other CAI courses are available to students which were loaned to CSU from East Texas State University.

A CAI Advisory Committee composed of nine faculty members and the Director of the Computer Center provides advice to CEL and discusses concerns related to program and utilization needs for the CAI Center. Two of the faculty members on the Advisory Committee are currently involved in using CAI courses.

#### The First Year

During the summer of 1976, CEL, through its instructional development grants, assisted four CSU faculty members enrollment in a ten day workshop on writing computer programs at the University of Akron. An objective of the workshop was to make faculty proficient enough to use the Coursewriter III Language so that they could not only design a CAI course but also be able to program the course into the computer at CSU. These same faculty members and a few others met regularly as a CEL Advisory Committee on CAI. Their efforts were the first evidence of a campus-wide CAI effort.

Under the direction of Dr. John Flower, Vice President for Academic Affairs, the Computer Center and CEL with the CAI Advisory Committee initiated the implementation of a CAI Center. The Computer Center installed six terminals connected by phone to the main computer and IBM's Coursewriter III system during the Fall Quarter of 1976. Two faculty members began writing CAI courses and quickly came to the realization that it was very difficult to find the time to design a CAI course and spend more time at becoming proficient programmers as well as providing the time required to program. Hence, CEL employed two CIS students to learn the Coursewriter Language and do the programming.

Recommendation 1: A workshop on CAI may be beneficial for fostering interest and providing CAI course design principles but if the workshop deals with learning a computer language it may not have practical benefits for all participants.

Recommendation 2: Provide computer programming personnel rather than having faculty do the programming.

By Winter Quarter, 1979 two CAI courses were being developed and programmed. During Spring Quarter, 1977 the CAI courses were ready for student use. One of the student programmers graduated and the other dropped out of

school. Another student was hired to learn the Coursewriter Language and perform programming assignments.

For the first 30 students who experienced the CAI courses, their experience was not entirely smooth going. A handout written by computer staff that explained how to operate the terminal and use the Coursewriter System was of little value to students who had never seen a computer terminal. Also, many students experienced delays of up to five minutes after input and then a response from the computer.

For the faculty members wishing reports on how their students performed on the computer, a utility program provided student responses that were saved on magnetic computer tape. However, the hard copy printout or these responses was voluminous and reported student responses over time rather than by student or by CAI course section. No short summary report was available for an instructor to determine at a glance what a student did or to look at a question and determine how many or which students answered it correctly.

For the new student programmer, a considerable amount of time was spent on trying to figure out what the other programmers had done. Consequently, when the faculty asked for changes, changes took much time. Also, the faculty were interested in getting certain reports summarizing information. This required, in some cases, considerably more programming to alter, delete, or change the programming that was done before.

Recommendation 3: Have something more than a handout for students to learn how to operate a terminal for CAI. Pictorials, a trained assistant nearby, step by step signs, a videotape showing steps, and a troubleshooting guide are some of the materials and measures that might be taken to insure students will be able to operate the terminals.

Recommendation 4: Pilot the CAI courses and hardware to debug any possible problems that might arise before students use the CAI course.

Recommendation 5: Have all programmer(s) document their programming with adequate explanations or comments on what operations and processes are being performed within sections of the CAI course. Save a hard copy printout of the CAI course.

Recommendation 6: Develop or purchase utility programs that will synthesize and summarize student responses into instructionally valuable reports.

Recommendation 7: Before any programming is undertaken decisions regarding information that the instructor might like the computer to provide in hard copy report and the purpose of the CAI course should be made.



**Recommendation 8:** If students are employed to write computer programs, hire those who are likely to be employed for at least one academic year. They should not only have had some programming experience but have as well, a reasonable command of the English language.

During the Spring Quarter, 1977 a new person was appointed to head CEL, and thus, in addition to providing instructional development for faculty, also coordinated the CAI Center and assisted faculty in developing and using CAI. A workshop through CEL was conducted for faculty wishing to learn more about CAI. The purpose of the workshop was to acquaint faculty with the CAI Center, get hands on experience using some of the existing CAI courses, and have participants learn how to apply flow charting in the design of CAI course software. Eight faculty members attended the workshop. However, after approximately two years since the workshop was offered, not one of the workshop participants had designed a CAI course to date.

**Recommendation 9:** A workshop to acquaint faculty with the operation of a CAI center may be beneficial, however, demonstrating how flow charting can assist a faculty member in designing CAI software may be premature in an introductory workshop session. Rather, a workshop that deals with how CAI can be used to assist, supplement, or aid a faculty member in solving some common teaching/learning problems may be more appropriate in establishing a need for CAI.

### The Second Year

During the summer quarter, 1977 a study of hardware problems relating to the extremely long response time was undertaken by the Computer Center and CEL made inquiries into existing software that might be borrowed from other institutions. It was determined that problems in delayed response time was due to the phone line connections as well as the nature of the six CRT terminals not being completely compatible with the computer. Because of these problems the six CRT's were replaced by four compatible CRT terminals and all terminals were hard wired directly to the main frame of the computer. As a result of these hardware changes, response time was reduced to seconds.

CEL's search to find existing and compatible software that might be loaned or purchased resulted in the addition of 15 CAI courses dealing with a variety of subject disciplines. The CAI courses were loaned to CSU from East Texas State University's CAI Center which also uses the Coursewriter System. The intention of securing CAI courses from other institutions was to defray the expense and time involved in having a CSU faculty

member design their own course. It was hoped that faculty would find the East Texas State CAI courses suitable for use as supplements to the subject courses they teach. However, in the 1 1/2 years since securing these CAI courses only one faculty member has used one of the courses as a direct supplement to his subject course. On the other hand, several faculty in the College of Education have had students experience several of these CAI courses to demonstrate the concept and applicability of CAI in education. The loaned courses have also been helpful to the Assistant Director of CEL in demonstrating to an interested faculty member some of the possibilities for CAI course development.

**Recommendation 10:** In selecting hardware (CRT-Terminals) to be used in conjunction with an existing university or college computer, test the compatibility of terminals with the computer to insure minimum computer response time.

**Recommendation 11:** If existing CAI courses are to be borrowed or purchased from other organizations: (1.) Check to be sure the CAI course language is compatible with your own (2.) Obtain a good description of the CAI courses before acquiring the programs and survey faculty to see whether the CAI course might be of value as it presently exists. (3.) A good reference for CAI course availability and course description is: Hoyer, Robert E. and Wang, Anastasia C., (eds). Index to Computer Based Learning, Educational Technology Publications, Englewood Cliffs, New Jersey, 1973.

Over the summer quarter, 1977 two faculty members approached CEL for instructional improvement grants to assist them in developing course material. Because of the nature of the project, it was suggested by the Assistant Director of CEL, that they develop a CAI course. They agreed and the CAI course was developed, piloted and tested for possible problems during the summer.

After the beginning of Fall Quarter, 1977, three CAI courses had been developed and were being implemented for student use. During the Fall Quarter CEL offered a workshop, at the request of the College of Education, to acquaint faculty with existing CAI courses and the CAI Center operation. A hands on demonstration was given for seven faculty members.

The CAI Center was operating fairly smoothly during the fall quarter, 1977, except students were still having some difficulties using the terminals. It was decided that perhaps the students might learn best by being taught while sitting at the terminal, thus learning by doing. Efforts were initiated to develop a CAI introductory course that would use the computer and terminal to teach them how to use the terminal. The course was named CAI.

During Winter Quarter, 1978, as a direct result of the Education College workshop, three additional faculty members utilized the borrowed East Texas State University CAI courses. The primary purpose in using these courses was to have future teachers, administrators, and counselors obtain exposure to the concept of CAI through a direct experience with it. By this time the CAI course called CAI, to teach students how to operate the terminal as well as giving them some insight as to the theory behind CAI, was available. The course was a success. Almost none of the students experiencing the introductory course required additional training or assistance in operating the terminal.

Recommendation 12: A hands on demonstration or a CAI course that explains how to use the terminal and allows hands on experience can be the most powerful and efficient means for insuring students learn how to operate the terminal.

Four other faculty members came to CEL for assistance in improving their courses. Through direct consultation between these faculty and the Assistant Director of CEL, it was determined CAI could offer a solution to their needs and be a suitable means for solving certain instructional problems. The development of three additional, faculty initiated, CAI courses were added by the end of Winter Quarter, 1978. Three CAI courses in addition to the other three developed at CSU were now available for student use for Spring Quarter, 1978.

By the end of Spring Quarter, 1978, four additional faculty were now using CAI. During this second year a total of 150 students from ten different subject courses had experienced CAI and eleven faculty were now familiar with the CAI Center.

Recommendation 13: The establishment of a CAI Center that bears some formal or informal organizational connection to an instructional development agency can be a beneficial arrangement in the application of CAI to instructional problem solving. Although faculty may not be aware of how CAI may solve instructional problems, the expertise of an instructional developer can demonstrate the benefits of CAI that faculty may not have otherwise realized.

CEL staff met with the CAI Advisory Committee and discussed some concerns. Namely, that although the growth of CAI on campus has been relatively slow in actual numbers of students using CAI, it was projected that because of increasing faculty interest, and because some of these faculty taught large lecture courses, that a scheduling problem might arise by Fall Quarter, 1978. It could be that with only four terminals not enough hours of terminal time would

be available during a specific time period, especially if several faculty wanted their students to have CAI courses during the same time. The recommendation of the committee was to have the Media Lab implement a scheduling plan.

The recommendation was followed and over the Summer Quarter, 1978 a packet of forms was developed for faculty to complete prior to having their students visit the CAI Center. From the information the faculty provided, schedules were posted in the Media Lab and students were permitted to sign up in advance of their CAI instruction. This procedure was tested during the Fall Quarter, 1978. One hundred and nine students that received CAI during the quarter experienced little difficulty in gaining access or operating the terminal.

Recommendation 14: An advisory committee to a CAI center can do much to assist in problem solving and the promotion of CAI center goals. Where possible, a committee composed of faculty and computer center personnel is desirable.

Recommendation 15: A formal procedure for collecting information on CAI course utilization by faculty and for scheduling student access to CAI terminals should be implemented to avoid any accessibility problems for students. Such a procedure is also useful in projecting terminal demand and insuring the student of a space at the terminal during a time that is convenient to him or her.

#### The Future and Additional Recommendations

As the CAI Center at Cleveland State University grows, it is helpful to reflect back on the experiences involved in software development. It becomes obvious from past experience, that CAI software development can be greatly facilitated by presenting a model or models to follow. Several models are described below. The intention of presenting these models, is to offer the faculty member wishing to design a CAI course, several helpful options or models to aid him/her in CAI course design. The models are presented by degree of complexity and individualization.

Recommendation 16: When assisting faculty in the design of CAI course software, it is helpful to present a model that can illustrate a course design format.

Model 1 shows how the design of a CAI course primarily centers around learning according to stimulus response theorists such as B.F. Skinner (1968) where the emphasis of reinforcement follows a student's response.

## Model 1

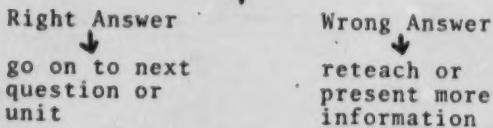
CAI program presents information/asks questions on information

↓  
Student answers question

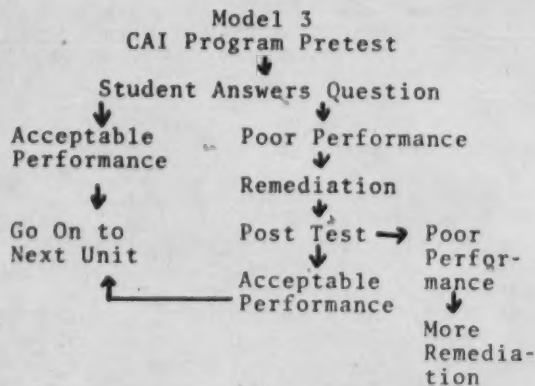
↓  
CAI program provides feedback (answer right or wrong)

Model 2 is closely aligned to programmed instruction techniques. Model 2 is chiefly different from Model 1 in that it allows for conditional branching. If a student does not answer correctly, he/she is branched to be retaught. If the student answers correctly, he/she goes on to the next topic, unit or question.

Model 2  
Same as Model 1

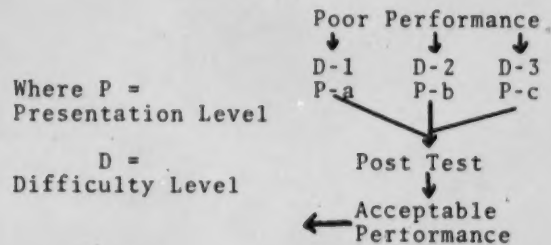


Model 3 is a more sophisticated version of Model 2 in that it is somewhat more individualized. Branching occurs on the basis of a student's performance on a pre-test. Further branching can occur if the student's post test score does not come up to expectations.



Model 4 contains features which have been commonly termed "response sensitive" (Kearsley, 1972). Branching in this situation is not pre-specified but depends to some extent on the pattern of the student's past performance. Branching is done to several presentation and difficulty levels, dependent upon the student's past performance.

Model 4  
Same as Model 3  
Except Poor Performance Branch  
as Follows:



## References:

1.) Kearsley, Greg P. "Some Conceptual Issues in Computer-Assisted Instruction," Journal of Computer-Based Instruction, 4 (1), August, 1977, 8-16.

(2.) Skinner, B.F. The Technology of Teaching. New York: Appleton-Century-Crofts, 1968.



FLORIDA'S STRATEGY FOR OPTIMIZING  
COMPUTING RESOURCES FOR  
EDUCATIONAL ADMINISTRATION  
AND INSTRUCTION

M. D. Roblyer & David Brittain  
Florida Department of Education

In 1977, the Florida Legislature funded the first year of a three-year effort to promote more effective, equitable use of computing resources in the Florida Educational System. The FECF\*, created as a catalytic staff to further this goal, adopted a "grass-roots" strategy consisting of several areas of activity, including: funding projects to encourage sharing existing computing resources with educational entities who are "have-nots" in terms of computing, establishing computing consortia to encourage sharing, providing technical assistance to school districts and colleges, negotiating state-level pricing agreements with vendors. This paper gives details of these and other activities related to Florida's approach to "equitable distribution."

\*Florida Educational Computing Project

### Introduction

It is a well-hidden truth of educational data processing that "computerizing" presents as many problems to state-level educational administration as it does solutions. There are continuing difficulties in assuring that: (1) all school districts and colleges - not just the large, affluent ones - have access to computer resources to facilitate their reporting and other administrative requirements, and that (2) hardware and software are being acquired and used to maximum benefit (à la Proposition 13). Since most local educational systems have a tradition of local autonomy in these matters, few states direct DP distribution from the state administrative level as does Minnesota, for example. Therefore, many other states may be interested in an alternative strategy which has been in progress for about two years in Florida and which shows promise for making DP available and cost-effective to all members of public educational systems.

The effort began in 1976, when the Florida Commissioner of Education was required by law to develop "a phased plan for distributing computer services equitably among all public schools and school districts in Florida." The first Phase of the plan, completed in February of 1977, proposed a way of dividing the State into a regional system of educational computing organizations and gave recommendations for methods of implementing an equitable distribution

plan. Phase II focused on specific recommendations for trying out the plan in two regions of the state, the South Florida and Central Florida areas. This pilot effort was to test and evaluate the feasibility of the basic premise of the proposed plan: that equitable distribution and optimization of computing resources could be achieved through sharing of existing capability on a voluntary basis. A key part of the Phase II plan was its expansion of the equitable distribution concept to include community colleges and universities.

Finally, these plans culminated in funding by the state legislature of the Florida Educational Computing Project (FECF). The intent was to establish a catalytic staff to provide technical assistance, and seed money to be used to support and encourage sharing activities. Based on the Phase I and Phase II Plans and the FECF's first year of activity (which is considered to be Phase III), a Phase IV Plan has been developed which, when implemented, will result in an equitable distribution of computing capabilities to all of Florida's Public Educational Systems.

The four-member FECF staff is charged with promoting equitable distribution through several key activities. Primary emphasis is always on encouraging a local, grass-roots commitment to sharing, rather than on directing such efforts from the state level. Although the



Legislature was specifically interested in the administrative area (financial accounting, inventory, personnel and payroll, etc.), projects of some limited kinds are also funded in the instructional computing area (computer management of instruction, guidance systems, etc.). In the following description, seven avenues of activity by the FECP staff and other state-level personnel will be outlined. As an additional aid to others who may be interested in implementing an FECP-like approach, some problems and unanticipated outcomes will also be discussed.

#### The Florida Equitable Distribution Strategy: Activities

The FECP is currently nearing the end of the second year of its three-year funding. As the function of the FECP was publicized throughout the State, it quickly became apparent that there was widespread demand and support for several areas of activity, not only to promote equitable distribution, but also to facilitate all areas of endeavor in educational computing in Florida.

#### Funding of Sharing Projects

Perhaps the most visible and important function served by the FECP is the identification of situations and applications which lend themselves to sharing projects to promote equitable distribution. Once a likely project is identified, the staff can assist in the development of the proposal, thus providing an opportunity to help influence the decisions that are made concerning hardware and software.

Not only were funds granted to the FECP, but the Division of Public Schools within the Department of Education also was supplied with funds to support the acquisition and sharing of common software. In the second year the Division of Community Colleges also received funds for this purpose. A project may be funded out of one source, or a combination of any of the three. The Legislature allocated the following amounts for each year:

|       | 1977-78   | 1978-79   | 1979-80             | 1980-81   |
|-------|-----------|-----------|---------------------|-----------|
|       |           |           | (Projected Amounts) |           |
| FECP  | \$530,000 | \$940,000 | \$940,000           | \$200,000 |
| DPS*  | \$595,000 | \$595,000 | \$595,000           | --        |
| DCC** | --        | \$250,000 | \$250,000           | --        |

\*Division of Public Schools

\*\*Division of Community Colleges

During the first year, only public educational entities were eligible to receive funding for such sharing projects. By the second year, the Legislature also elected to include local governmental agencies as possible participants in

sharing projects. Most projects funded during these years were administrative in nature. However, there were some instructionally-oriented activities as well. All funds were distributed on a contract basis, rather than as grants. In each case, funds were disbursed via a contract between the State of Florida Department of Education and a school district, community college or university acting as the fiscal agent. In addition to identifying and coordinating proposal activities, FECP staff also write the contracts and monitor their progress throughout the life of the project.

Three kinds of projects were perceived to fall under the "sharing" heading. Fundable activities included the following:

1. Modification/sharing projects - These activities are the major focus of educational computing funds. A member of the educational system may be funded for the costs and expenses involved in providing access to its data processing resources by another member in the system in order to meet a demonstrated need. The submitter of a sharing proposal should be able to show evidence of the need for the capability, how a sharing arrangement will satisfactorily meet that need, and why this method is a cost-effective and/or appropriate solution. For such projects, funding may cover the costs of modifying software in order to allow it to be run at another site, technical assistance and consulting required for modification and implementation activities, and/or hardware such as terminals and communications equipment required to allow a site access to a data processing center.
2. Acquisition/sharing projects - In some instances, there is a need to acquire additional computing capability in order to provide a needed service to one or more members in the educational system. Fundable activities under this category include hardware upgrades to providers of computing capabilities in order to allow continued service, and software purchases in situations where such acquisitions have high potential for long-range benefits to the state in terms of (1) cost savings and/or cost avoidance, and (2) in terms of promoting the concept of equitable distribution in the future. Submitters of proposals under this category should be prepared to show how an acquisition of hardware or software can accomplish benefits of these kinds.

3. Pilot/sharing projects - A small proportion of the available educational computing funds will be directed toward establishing the applicability and/or cost-effectiveness of new technology to educational problem areas in Florida. Fundable activities include projects in

**348** instructional computing which have a primary focus of making available computerized products to facilitate the teaching/learning process in Florida schools. Proposals in this category should include a description of the educational problem area which is being addressed, and the way in which a computerized product may offer unique capability to meet needs in this area.

The intent of the Legislature and the wording of the proviso language which established these funds allows for some degree of interpretation as to what constitutes "promoting equitable distribution." However, some activities are clearly contradictory to the purpose of the funding. Since the primary focus is to pursue cost-effective activities to optimize available resources whenever possible, development activities are not fundable under these categories of money. Any activities which would constitute a proliferation of resources at a given site is also counter to the intent of the funding, as is establishing a large staff of persons who may not be able to continue on local funds after state funds are no longer available. It is also important to note that "saving money" cannot be the primary objective of a project. There must be some real evidence of needs before a sharing, pilot, or acquisition project is proposed.

#### Evaluation of Hardware and Software Acquisition

The legislation which funded the FECP also contained a provision requiring the Commissioner of Education to review and approve all purchases or leases of hardware or software over \$6,000 in a 12-month period. The FECP serves as the body which supplies this review/recommendation to the Commissioner's office. A procedure has been set up in which an educational entity fills out a form identifying the equipment and/or software to be acquired and gives a brief rationale for the need. The staff has endeavored to provide rapid turn-around on these reviews so as not to impede progress in any organization. This, again, is an interface point at which technical assistance and direction can be supplied.

#### Providing Technical Assistance

As the emergence of computing consulting firms has indicated, there is a growing need for technical assistance to all organizations, educational or otherwise, which are not able to hire a full-time staff to carry out such activities. Assistance takes several forms. An important function is aid to school districts and colleges in the preparation of bid specifications and the evaluation of hardware/software proposals from

vendors. Another area is in the evaluation of data processing center facilities and procedures in order to determine ways of optimizing existing resources and to plan for change. Still another area is the use of study teams to assist in determining viable directions for future acquisition of hardware and software. Finally, technical assistance on the selection and use of instructional computing applications and resources is supplied.

#### Supporting Consortium Sharing Activities

One of the primary instruments in promoting the sharing of computing applications is an organizational structure which facilitates communication among potential sharing partners. The first regional groups for consortia were in the southeast (Miami) and central (Orlando) regions of the State. These areas were the first to be funded for sharing activities, and were leaders in organizing functional computing consortia. Other regions of the state soon followed this example and organized computing consortia consisting of the member institutions and school districts in that geographical region (See Figure ). There are eight consortia at this time (seven are regional and one is vendor oriented), and the members report that the organizations are useful in assessing member requirements and in generating ideas for proposals to make use of the funds available from the state.

#### Negotiating Hardware/Software Agreements

The FECP staff has found that a significant role which they play in educational data processing is that of maintaining communication with vendors of hardware and software. This communication provides an opportunity to keep up with the state-of-the-art in important areas of change, as well as to identify and solve problems in regard to purchases by educational entities. As a state agency, the FECP can also speak on behalf of the Department of Education in negotiating discount agreements for hardware and software acquisitions for members of the educational system. The agreements which have been reached to this date have resulted in substantial savings and benefits to school districts, community colleges, and universities throughout the state.

#### FECP Newsletter

One method of facilitating communications among members of a consortium and among consortia in the state is a small, monthly newsletter with several central features of interest to educational data processing personnel. These include a "Buy/Sell/Swap" column which lists equipment available for trading or selling, as well as equipment which is desired to purchase or lease. Projects funded out of

## FLORIDA'S REGIONAL COMPUTING CONSORTIA



NOTE: Educational Computing Consortia include all school districts, community colleges and universities in a geographic region.

- |          |                                                       |
|----------|-------------------------------------------------------|
| Region 1 | Gateway Educational Computing Cooperative             |
| Region 2 | Northeast Florida Educational Computing Consortium    |
| Region 3 | Central Florida Educational Computing Consortium      |
| Region 4 | West Central Florida Educational Computing Consortium |
| Region 5 | Big Lake Educational Computing Consortium             |
| Region 6 | Southwest Florida Educational Computing Consortium    |
| Region 7 | Southeast Florida Educational Computing Consortium    |



**350** common applications monies are also described, as are negotiated agreements with vendors. The newsletter has proven a valuable instrument to assure that everyone is aware of significant events in educational data processing at both the state and local levels.

#### Promoting Computer Literacy

Since educational administrators can only make sound decisions about needed computing capabilities and potential for sharing computing resources if they are aware of the technology and what it can do for them, the FECF staff also provides technical assistance in the form of promoting computer literacy. Although this is done on a daily basis in the form of explanation and advice on various issues, another vehicle for this training is the FECF Seminar in Computer Literacy, a traveling one-day show put on by request in school districts, community colleges, or other educational organizations. The seminar is the product of a systematic instructional design process, and has been formatively evaluated and modified in order to better meet the needs of educational administrators and instructors in Florida.

#### The Florida Equitable Distribution Strategy: Problems and Concerns

During the first two years of the plan's implementation, about \$3,778,590 have been realized in cost savings and cost avoidance due to sharing and optimization of resources in the state. There is evidence that significant progress has been made toward an equitable distribution of computing capability. However, other states who wish to implement a similar strategy should take note of certain trade-offs in using this method. While there are a great many benefits to be derived from a grass-roots approach to bringing about change, it is also true that persuasion takes longer than direction. Innovations such as using and sharing computing capability have more chance of taking hold and flourishing if there is support from local levels for such activities. Such support, however, takes time to generate. There are also problems in overcoming hardware and software incompatibility and parochial suspicious and interests in order to bring about a given sharing activity. Such problems would, of course, not exist if state-level administration dictated a common vendor for all members of the educational system. On the other hand, it is quite possible (even probable) that local organizations and personnel would vehemently resist such change if it were approached in this way.

A great deal of the success in the efforts to date must be attributed to close communication with and individual attention to members of the educational community. This was not accomplished over the telephone alone. The small FECF staff must maintain personal contact with data processing and instructional personnel in 67 school districts, 28 community colleges, nine universities, as well as the regional data centers and vocational schools. As one might suspect, considerable travel around the state is involved. Knowing by heart the daily schedules of Air Florida and Southern Airlines has become almost a prerequisite of employment for FECF staff. Again, this communication is crucial--but time-consuming and taxing.

#### Conclusion

Although it was initially funded for only three years, it has become apparent that the functions of the FECF staff have played an important role in educational data processing in Florida. It is anticipated that the staff will be supported at least one additional year, and the possibility exists of its becoming a permanent structure in the Department of Education. Part of the reason for this is that equitable distribution on a grass-roots basis can not be accomplished in just three years. In addition, personnel in the state have discovered coordination and communication needs they never knew they had. The future of educational data processing depends heavily on careful planning and on the generation of commitment to effective DP at all levels of educational administration. We in the FECF feel that Florida's approach to equitable distribution can be a successful method for accomplishing these things in other State systems.



## FROM TIMESHARING TO MICROCOMPUTERS

Donald C. Holznagel  
TIES

The factors of increasing telecommunications costs, aging timesharing systems, and the increasing capabilities and decreasing cost of microcomputers has caused the TIES regional computer center to draw an evolutionary plan for change. The rationale, plan, and practical aspects of its implementation are discussed.

Introduction

The Minnesota School Districts Data Processing Joint Board (TIES) is a regional consortium of 55 school districts for the purpose of providing instructional and administrative computing services to its members. Over 100 secondary and 200 elementary schools and 260,000 students are contained in those districts.

Instructional computing service began in December 1969 with a 16-port Hewlett-Packard time sharing system, expanding to three 32-port HP systems by 1974. In addition, purchase of time on the state-wide timesharing system has reached 84 port equivalents by the end of 1978. That system is operated by MECC, the Minnesota Educational Computing Consortium. Hence, the total timesharing capability provided from the TIES budget to the member districts now is 180 ports.

Several factors, taken together, have caused TIES staff members to examine the instructional computing needs of member districts and several alternatives for meeting those needs. The Hewlett-Packard (HP) systems at TIES are nearing the end of their useful life, telecommunications costs are rising, and microcomputers are increasingly available at rapidly decreasing costs. Nevertheless, a continuing need for timesharing service exists.

School district staffs are growing in appreciation of the value of computers in the instructional process while beset with declining enrollments, teacher layoffs and taxpayer resistance to new expenses. Member districts are looking to TIES staff for advice and direction. A planned implementation of new hardware over a period of time is preferable to an unplanned transition, because it allows for better budget and staff use planning both at TIES and in member districts.

Consideration of these factors has resulted in the delineation of the alternatives discussed in this paper, and the selection of one of them.

Rationale

1. TIES current HP systems are outmoded and support is diminishing.

The 2000F and 2000C' systems are no longer manufactured or sold by HP, and maintenance costs are gradually increasing as support (parts, enhancements, etc.) is declining. After April 1, 1979, HP will no longer offer a maintenance contract on the two 2000C' systems. Thus, other methods, including maintenance on a time and materials basis and acquisition of a parts inventory, must be employed. The failure frequency in the past year has caused us to question the future reliability of the systems. In addition, the F/C' version of BASIC is surpassed by current versions in standard use on other systems, including most microcomputers. Furthermore, the system software does not provide a sufficient amount of information to monitor hardware and software use adequately, either from a staff or user standpoint.

2. Communications costs are expected to rise.

Increases in the cost of a standard business phone line have already been experienced. In addition, we expect the price structure for telephone service to change. This might occur by reductions in the toll-free dialing area, introduction of usage-sensitive billing, or a combination of those or other methods. Any such action will tend to increase user costs. We cannot predict the dates of implementation of such plans, but since they are

already in use in other metropolitan areas, we can reasonably expect them in the Minneapolis-St. Paul area.

3. Microcomputers having extensive capabilities are available at low cost when compared to timesharing systems.

Current microcomputer systems have cost advantages over timesharing systems for a number of major applications such as programming instruction. For example, two small microcomputers, with BASIC superior to that of the HP 2000 series, can be obtained for the price of one timesharing terminal. On the other hand, one microcomputer with sufficient main memory and mass storage to process most timesharing applications is available at approximately the same cost as a timesharing terminal.

Microcomputers expected in the next few years will have increased capabilities at decreased costs, with increased advantages over timesharing systems for classroom applications. Color, graphics, and inexpensive peripheral equipment for such things as speech synthesis, already make microcomputers very attractive for many instructional applications and for curricular areas, such as special education and music, which are as yet relatively unaffected by the computer.

Our initial experience indicates that acquisition of microcomputers is for added computing power and new users. However, timesharing port replacement can be expected and will gradually be phased-in.

4. There is a continuing need for timesharing systems.

Nearly 600 terminals are currently in use in TIES schools. These will continue to be used until they can be gradually phased-out. Thus, the need for timesharing exists, especially during the transition period.

Listed below are areas of interest and use which require continued timesharing support and availability:

- A. Increased interest and activity in the use of the HP Instructional Dialogue Facility indicates a continued need for tutorial applications and a simple authoring facility. In addition, the HP Elementary Mathematics Drill and Practice package has had a gradual increase in use. Interest exists in drill programs for other basic skills, particularly language arts, and for students with special needs and learning

disabilities.

- B. There is an increasing use of student guidance support programs such as the Guidance Information System and the Minnesota Occupational Information System. Both systems are desired. Terminals have been acquired through district, state, and federal funds, specifically for the use of these program packages.
- C. Other large information retrieval programs and application packages such as COMPUTE and INQUIR, which are not now appropriate for microcomputers because of program size or language, require a timesharing system.
- D. Interest has been demonstrated in FORTRAN, COBOL and other languages in addition to BASIC, for student programming and certain applications.
- E. A central timesharing system is needed as a library and distribution source for microcomputer programs.
5. Continued TIES staff support of instructional computing in member districts is required.

Such support now includes teacher inservice workshops, user assistance by phone and on site, development of programs and instructional materials, and maintenance and documentation of a program library. The increasing use of microcomputers implies additional support needs in the same areas but related to a variety of hardware. A telephone survey of member district coordinators indicated a desire for continued TIES support of a program library, training and on-site assistance for microcomputers. A questionnaire sent to teacher personnel revealed application development, teacher inservice and classroom assistance as the three most often mentioned expectations of TIES staff.

#### Assumptions

The following assumptions were made in formulating the various alternatives for meeting the instructional computing resource needs of the TIES member districts.

1. That microcomputers will provide port equivalent capabilities to replace many instructional computing applications currently provided on timesharing systems.
2. That projected growth of instructional usage which requires additional computing resources will be handled by the

acquisition of microcomputers beginning September 1, 1979.

3. That the long range projection of timesharing needs is 96 ports.
4. That the mode of distribution of program packages will be down-loading from the timesharing system to the microcomputer.
5. That all currently used modes of instructional computing applications will be provided to users.
6. That Design & Development and Services support for microcomputers and application programs will be provided by TIES.
7. That staff resources from MECC will be a supplement to TIES levels of staffing.

#### Description of Alternatives

Five alternative plans for providing instructional computing resources were reviewed by TIES staff. Each of the plans was based on the objective of eventual dependence upon microcomputers for the majority of instructional computer applications. Figure I displays the number of timesharing ports to be provided in each year under each plan. Where two plans share the same pattern, the difference between them lies in the expected source of the timesharing ports. Plan 2, for example, assumes that TIES would retain 96 HP ports through June, 1980, reducing by 32 ports per year for three years thereafter. Plan 3, while maintaining the same total number of ports, assumes that reduction of HP ports begins in July, 1979, but that additional ports are obtained from MECC to make up the total.

A detailed presentation of alternative 2, which was finally chosen, is shown in figure II. It implies that member districts must acquire microcomputers equivalent to the service provided by 32 ports by September, 1980, to replace the port reduction. If growth in computing is expected, then additional micros will have to be purchased. Since we are now serving districts at the ratio of 3 terminals per port, then port replacement alone would require that 96 microcomputers be purchased by the districts each year for 3 years, an average of two micros per district.

Alternative 1 proposes an almost immediate switch of all HP use to micros. The timeline implied in Alternative 1 for hardware acquisition by member districts and staff work at TIES appears unrealistic. Acquisition of so much hardware in so short a time may raise problems in building and justifying a budget in a

school district. The timeframe for program conversion, software development, and planning and conducting user training sessions by TIES staff is too short. Hence, this alternative is impractical both for the districts and TIES, and was not considered further. 353

Alternative 5 proposes to replace the HP systems with new replacement timesharing systems. That would cause TIES to install new hardware having a useful life of seven years, at a time when such hardware is becoming less attractive for several reasons discussed in the rationale. It is counter to the statewide plan and would not likely receive approval. Therefore, this alternative was also not recommended.

Alternative 2 appears to offer a reasonable time for districts to plan and budget for micro acquisition, and for TIES staff to plan and conduct inservice sessions and convert programs. It also does not stretch the limits of maintenance to an untenable degree, although a supply of spare parts will be needed on site. Finally, the port requirement from MECC is more even year to year, enabling their staff to plan without wide fluctuations in ports to be delivered.

#### The Microcomputer Aspect

One of the important outside factors on which the plan depends is the state contract for microcomputer acquisition by educational institutions. A MECC task force was formed early in 1978 to define the educational requirements for microcomputers. The task force recommended the identification of a single microcomputer for instructional use in schools. The result was a bid specification issued to vendors, which in turn resulted in a contract award to Apple II. The agreement allows any educational institution in Minnesota to obtain an Apple II micro at an advantageous price. This enables TIES districts to move into the port replacement aspects of our plan more easily. It also enables TIES staff to concentrate development and inservice efforts on one machine. Providing the support districts want and need would be virtually impossible if there were a wide variety of micros in use.

In addition, through the bid process, the features and capacities required for effective use of existing applications and new development were specified, and became the statewide standard. Enough main storage was obtained in the standard machine to be the equivalent of the new space available on the HP timesharing system. Hence, the task of converting the program library to the micro has been greatly facilitated.

9/18/78



Ty79-Ty84

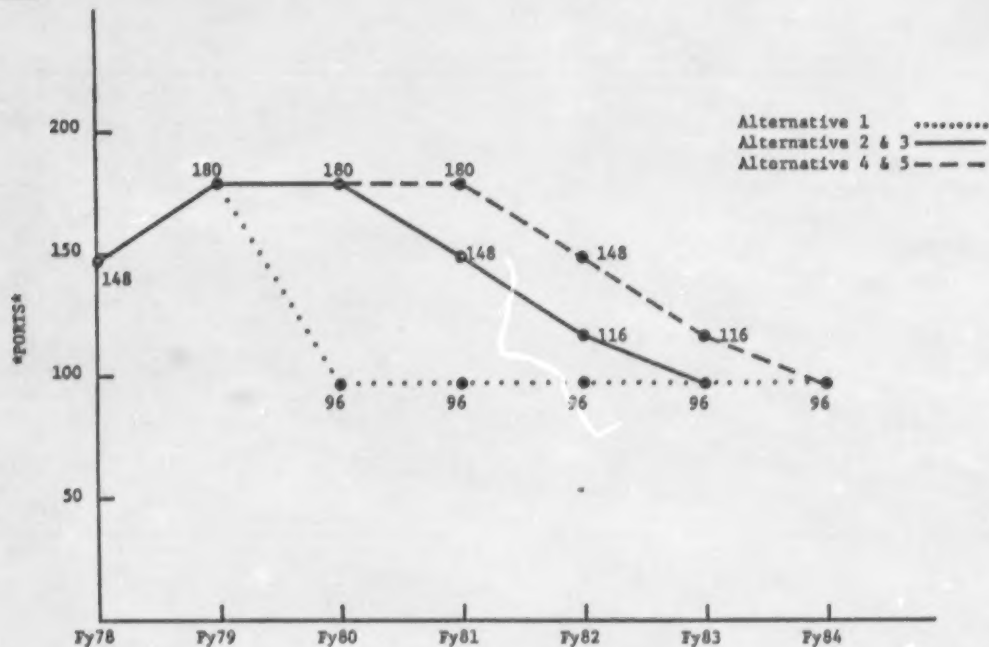


Figure I

\*YEARS\*

INSTRUCTIONAL COMPUTER PLANNING

ALTERNATIVE 2

9/18/78

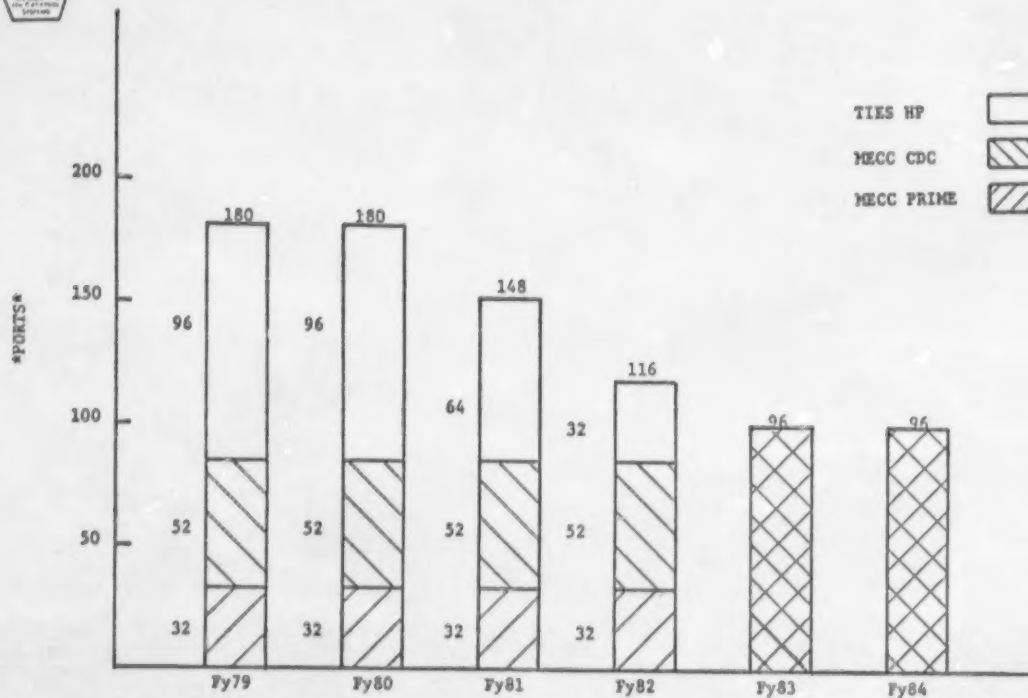


Figure II

\*YEARS\*



### The Status of the Plan

We are now six months into executing the plan since its approval by the TIES Board. Our goal is to have most of the timesharing library converted to the Apple II and available for downloading from the statewide timesharing system by users by September, 1979.

The downloading (and uploading) program has been completed by MECC staff. A user needs only to log in to the timesharing system, using the Apple as a terminal, and execute the download program. Application programs in Apple II BASIC code are stored in a special library, from which the user can select any program by name, transferring it to the mini-flexible disc storage attached to the micro.

The TIES library has been converted to the statewide timesharing system, enabling users to access the programs on either system. The process of converting the library programs to Apple code is underway and is expected to be completed by September. **355**

Inservice training has begun with familiarization sessions for teachers and administrators. BASIC language sessions have been designed. Individual assistance when a machine is delivered is being provided. Approximately 50 micros have been ordered, with deliveries occurring monthly. District response to the plan and its implications for them has been enthusiastic.

IMPLEMENTATION  
of the APPLE II MICROCOMPUTER  
in MINNESOTA

by Linda Borry and John Arneson  
Instructional Coordinators  
Minnesota Educational  
Computing Consortium

This paper describes Minnesota's efforts to implement the APPLE II microcomputer into schools throughout the state. Points which will be addressed include: status of instructional computing in Minnesota prior to the advent of microcomputers; the microcomputer study; the bid for a state contract for microcomputers; teacher inservice; applications software development, acquisition, and distribution; and demonstration of selected courseware.

#### Purpose and Intended Users

The Minnesota Educational Computing Consortium (MECC) presently provides instructional timesharing service to 75 percent of the elementary, secondary, vocational school districts in Minnesota as well as to the higher education institutions in the state. This service is provided on a 375 port Control Data CYBER 73 system, and a 50 port PRIME system. Delivery of this service necessitates a very extensive telecommunications network.

During the 1977-78 school year it became evident that microcomputers had evolved to the stage where they were packaged to be used by non-technical people; had a good BASIC Language; and could compete price-wise with other computing resources.

Minnesota teachers began to show interest in microcomputers and their classroom potential. Commodore PETs, Radio Shack TRS-80's and others started to appear in schools. MECC decided to take an "actionary" rather than reactionary approach to the introduction of microcomputers in Minnesota schools.

During the '77-'78 school year, a microcomputer task force was organized to: study and evaluate the hardware and software capabilities of existing microcomputers; determine which types of applications presently being done on the timeshare systems could be done on a microcomputer; and determine the roles of MECC, its member systems, and individual institutions in the acquisition, support, and implementation of microcomputers.

#### Method

The Microcomputer Task Force produced an extensive report which included the following:

- 1) Evaluation of hardware, systems software and applications software on 14 microcomputers.
- 2) Definition of a minimal system for education.
- 3) An Invitation For Bid for a state contract enabling volume purchasing of microcomputers for educational use.
- 4) Definition of the roles of MECC, its member systems and individual institutions in the implementation of microcomputers.

The Microcomputer Task Force Report was published and made available for purchase. MECC is now in the process of updating the report.

The Invitation For Bid was released; bids were received; and APPLE was awarded the one-year contract. On October 16th, 1978, schools could begin buying APPLES at substantial cost reduction.

When the bid was released, it was anticipated that schools would order approximately 10 microcomputers per month. By March 1st, four and one half months after the awarding of the bid, almost 200 APPLES had been ordered.

The concept of a state contract not only provided a vehicle for acquisition of

microcomputers, but also simplified the question of MECC's role in their support. MECC's efforts could now be focused on one single system rather than on a myriad of different ones.

MECC staff gives full support to the state contract (APPLE II) microcomputer. This includes acquisition, inservice, documentation, problem solving, and curriculum development and dissemination.

MECC serves as an interface between APPLE and the users, and as such, handles all user orders and all communications with APPLE. When systems are shipped, MECC verifies orders and arranges for delivery to districts.

The MECC instructional coordinators have become familiar with APPLE and have conducted hundreds of demonstration and inservice sessions for users throughout the state.

Although APPLE documentation is of very high quality, MECC staff have provided additional documentation such as: unpacking instructions, a new user's guide, and guides to graphics, sound, and files.

MECC has a full time consultant available to users throughout the state by means of an IN-WATS line. The consultant, in addition to answering timeshare questions is also addressing APPLE questions.

MECC coordinators have been involved both in converting programs from the CYBER to the APPLE and in developing new programs which exploit the unique capabilities of the APPLE, namely: speed of display, graphics, and sound.

Perhaps the most exciting aspect of the APPLE implementation is the method for dissemination of software. One of the bid requirements was that the system must have an RS-232 interface. This interface is the key element in the software dissemination effort, as it allows users to down-load programs from the CYBER system to the APPLE. MECC has a library of 75 programs written for the APPLE which are stored on the CYBER system. Using their APPLES as terminals, users can call the timeshare system and request that certain programs be sent to their APPLE microcomputers. This capability eliminates the need to copy disks, mail them, etc.

#### Results

It is very difficult to objectively assess the results of Minnesota's great APPLE project, as the project is still in progress. Four results which can be

discussed, however, are: the increase in microcomputer literacy among teachers, students, and administrators in Minnesota; the comparatively smooth transition from a strict timeshare environment to a mixed the comparatively smooth transition from a strict timeshare environment to a mixed timeshare/microcomputer environment; the accomplishment of having gone through the process of preparing a bid, acquisition, inservice, and curriculum development and distribution; and the definition of an instructional system. 357

Microcomputer literacy among Minnesota teachers, administrators, and students has increased significantly this year. For Minnesotans, microcomputers have changed in image from something seen in a magazine or on television to something found in the school or home. The impact of the microcomputer and its potential have fostered a tremendous renewal of interest in the whole field of instructional computing.

Microcomputers are now playing and will continue to play an increasingly important role in instructional computing. During these years of transition from a strict timesharing environment to an environment which is a mix of timesharing and microcomputers, it is important to define how each resource can best be used. These two resources should be regarded as complementary rather than competitive. We feel that Minnesota's approach has promoted this view.

The microcomputer world is anything but static. It is virtually impossible to keep up with all the latest developments and equipment. MECC is aware of this and the implications that it has for MECC's continuing involvement with the APPLE II microcomputer. As time goes on, MECC may very well want to extend its support to other brands of microcomputers. Having experienced the implementation process has made MECC "older and wiser". Starting to support other microcomputers will be much easier as a result of experiences with the APPLE.

A fourth very important, but often overlooked result is that through its study and Invitation For Bid, MECC has defined an educational microcomputer system. Vendors who are interested in competing in the educational market have been informed about the needs of education. This is a very positive result for both the vendor and the educator. Vendors can tailor their systems to the needs of education, and educators will have appropriate systems from which to choose.

MECC has taken a positive step to integrate microcomputers into the instructional computing scene in Minnesota by its support of the APPLE II microcomputer. Schools in this state can purchase microcomputers at a substantial cost reduction, receive support and inservice, and have access to a library of quality educational applications.

#### Descriptions of Microcomputer Software to be Demonstrated

The MECC instructional computing staff has produced some major instructional packages during the short time that they have had access to the APPLE II microcomputer. These instructional learning packages focus on several objectives:

- 1) Using learning activities where a graphics display adds to the motivational impact of the activity.
- 2) Using learning activities that require minimal text for ease of use.
- 3) Providing written documentation which includes the educational intent of the learning package.
- 4) Using learning activities where the audio expression adds to the motivational impact of the activity.
- 5) Using learning activities that are suitable to the tutorial, and drill and practice, and simulation modes of use.

Four of the completed packages are briefly described as follows:

#### MUSIC THEORY

This series of programs drill on all aspects of music theory. They include excellent graphics and audio sound to motivate students in recognizing such items, for example, as missing notes or wrong notes. The student is presented with a colored staff in the graphics display, as well as the musical tones of the notes.

#### COORDINATE GEOMETRY

This series of programs display a tutorial of how conic sections are produced using angles and line segments. They employ a tremendously imaginative and colorful display to produce such figures as astroids, circles, quadrifoliums, and hyperboloids, just to mention a few.

#### DRIVER EDUCATION

This series of programs generates questions on the skills and knowledge required of a teenager preparing for a driving test. The multiple choice questions include a graphics display of the road signs as they would appear in an actual driving test. These programs were converted from the MECC timeshare library and have always been popular. Now with the added graphics they will be even more so.

#### PLANE GEOMETRY

This series of programs is a tutorial and drill on beginning concepts of geometry. In order to understand how lines, angles, and triangles are formed, a basic understanding of points, line segments, rays and vertices is necessary. These programs with their graphics display show how basic concepts provide the necessary ingredients to build geometrical figures.



"INSTITUTIONAL PLANNERS AND  
COMPUTER CENTER ADMINISTRATORS:  
NEED FOR GREATER DIALOGUE"

Robert F. Garnett, Ph.D.  
Office of Institutional Planning

Dianne Garnett  
Asst. Director, Computer Center

University of Evansville  
Evansville, Indiana

Institutions of higher education in ever increasing numbers are devoting substantial energies and resources to institutional planning. Allowing for uniqueness of institutional administration and governance, a variety of approaches have proven successful; however regardless of the approach chosen the effectiveness with which the administrators responsible for the planning function interface with the administrators responsible for the data process function will greatly impact the overall outcome of the planning effort. This key relationship is taken for granted and, perhaps, at times simply overlooked. The presentation will address this relationship with focus on the establishment or modification of a management information system as a parallel step in implementing a comprehensive institution-wide planning effort.

At the national CAUSE Conference in New Orleans in December, 1978, L. Sharon Topping, Coordinator of Management System Design for the State University System for Florida made a rather interesting presentation, "Let's Stop Blaming MIS." The most significant conclusion to be drawn from this presentation was reinforcement of something most of us have learned from personal experience, but nonetheless, our behavior does not demonstrate that we have in fact learned. Their experience in Florida dramatized the fact that the major obstacles in implementing any Management Information System (Planning System) were not technological in nature (hardware, software) but rather the human factors. The Florida experience as related by Sharon Topping made it very clear that the best designed system will never function properly, if at all, unless good communication occurs on a regular basis between all of the key personnel involved, and secondly that procedures and policies are modified accordingly. When the latter does not occur the system performs sluggishly at best and perhaps not at all. Miscommunication compounds rather quickly and before long the system has developed an image which lacks credibility and may in fact fall into a state of disuse, with most unit administrators attempting to return to the old way of doing things. The sad part of all of this is that the institution has lost a substantial degree of its economic investment in hardware as well as human resources and created an atmosphere of both frustration and confusion.

Within that context, it is our intention to focus on what has been termed a "Key Relationship," that is, a relationship between those responsible for institutional planning and those responsible for administration of the data processing on an institution's campus. Whether an institution is embarking on the planning effort from scratch or whether they are attempting to modify an existing system that has been developed over a period of years, the fact remains that these two administrative units must work hand-in-hand with clearly established lines of responsibility. It is safe to say that in 1979 more institutions are in the process of modifying systems that have developed in somewhat a fragmented fashion over a period of years rather than implementing an initial system. As a consequence, the task is even more complicated since a typical system is the result of input and modification by variety of changing personnel over a period of years. The challenge of modifying an existing system without impairing the continued operation of the institution is a challenge indeed, and therefore, catapults the responsibilities of the Computer Center's staff immediately into a dual role of maintaining operations at an acceptable level while modifying and implementing a new system concurrently.

Why is this such a KEY RELATIONSHIP and how will a successful partnership once developed have positive impact not only on a Management Information System per se, but on the Institutional Planning effort in general?

**Institutional Administrative Structure:** In higher education the Office of Institutional Planning and Director of Computer Center operations are typically responsible to different offices i.e., Vice Presidential level or directly to the President. Only in those institutions having an Executive Vice President will you likely find these two offices responsible to the same person. The purpose of this paper is not to argue the pro and con of having the institutional planner and director of Computer Center responsible to the same office, on the contrary it is to recognize the reality that they typically do not and consequently additional efforts must be made to insure the highest level possible of sound communication and collaboration.

**Traditional vs. Emerging Role of Computer Centers:** While Computer Centers are established on most campuses with administrative purposes in mind, they have in the past two decades become heavily involved in academic computing, academic research as well as more recently institutional research. Moreover, many Computer Centers provide a selected amount of service to the education and business community.

These changes have great implications philosophically as well as operationally and highlights such questions as the following:

1. When, who, and how should institutional priorities be established for Computer Center operation? We have long since passed that point when most Computer Center operations (hardware and staff) were greatly underutilized; when Computer Center directors were eager to accept any and all requests from administrative units, faculty members or the community to more fully utilize the current facilities. The fact is that we have now arrived at the point where the demand for computer services are equal to or greater than the ability of the staff along with available hardware and software to meet the needs. As a consequence, the question of priorities has come into focus quite sharply.

Precisely what portion of the Computer Center time should be devoted to administrative functions? What percentage to academic computing? What percentage to research? What percentage, if any, to community service, that is, external activities which are income producing.

2. For those institutions who have chosen to modify or establish a

Management Information System to serve the institutional planning process, what is the role of the Computer Centers administrators in the establishment and implementation of such a system. While the literature frequently addresses the need for wide-spread involvement of middle-management administrators, it appears that less than adequate attention is given to the crucial role of the Computer Center staff. ARE THEY A SERVICE BUREAU OR ARE THEY AN INTEGRAL PART OF A MANAGEMENT TEAM?

**Comprehensive Planning:** During the past twenty-five years most all institutions of higher education have engaged in some degree of planning as well as developing some form of "Information System." The phenomenal growth rate experienced by most all institutions of higher education during this period necessitated increased attention to the areas of planning particularly facilities planning and to a lesser degree academic and financial planning. Without question this boom gave an enormous boost to the successful entry of data processing into higher education. While there is no need to dwell on what is now history it is important for our purposes here to make two observations concerning both the planning function and the role of data processing in the "Information System." Planning was for the most part an organizationally discrete function except at the top level of decision making, i.e. the chief executive officer and the Board of Trustees. Discrete in the sense that the focus was on academic planning, financial planning, facilities planning, etc. and while there was some attention given by those responsible for each of the areas to the other there certainly was no mandate that the areas be coordinated and relate to one another. Data processing during this period of time grew rapidly to the powerful position it holds today. Institutions today are in fact data processing dependent. We may not enjoy acknowledging such, but the fact remains. It is regrettable to note that the information system developed during this era of growth was quite fragmented. Moreover, there was no formal relationship between the information system and the planning function.

That period of affluence has passed and today administrative personnel in institutions of higher education are faced with an ever increased number of complex challenges and responsibilities; therefore the need and desirability for comprehensive institutional planning have become apparent. Administrative decision making in a period of affluence is costly, but frequently can be absorbed with little notice and certainly no dire effect upon the institution. The converse is true today when the institutions, public and private alike, are faced with declining

enrollments, declining sources of federal, state, foundation and corporate fund and simultaneous with the impact of inflation and maintaining quality instruction. The margin of error in many institutions today is such that faulty judgements and poor planning can lead not only to deficit, but in fact dissolution of the institution itself. I would hasten to add that survival is not an end in itself for the majority of institutions in higher education. Survival, yes, providing the institution can remain viable and maintain a quality program in keeping with its mission.

For these reasons and others, institutions have moved toward a comprehensive approach to institutional planning to insure that planning in all areas be it academic, facilities, financial or development are closely coordinated. At the same time administrators have come to recognize that the precision required in planning today demands easily accessible and reliable data to assist in decision making process at all levels. Consequently, the need for a Management Information System that is uniquely designed to meet the needs of the planning function is mandatory.

At the University of Evansville, where the authors are employed, the responsibility for development and implementation of both the mechanism for institution-wide planning as well as review and modification of the Management Information System has been given to the Office of Institutional Planning. The approach to planning at the University of Evansville is certainly not dissimilar to other institutions. The approach taken was essentially an organizational approach involving all "cost centers" where the managers of these cost centers may carry the title director, supervisor, department chairman, dean, or vice president. The primary objective of the initial year was that of assessment beginning at the unit level (department chairman) and succeeding upward through deans or directors to the respective vice presidents (academic affairs, administration, student affairs, development.) The focus was on the present in terms of perceived mission, priority objectives, assessment of programs and services rendered and an analysis by each unit administrator of the resources available to them (personnel, facilities, support services, budget) to carry out its perceived mission and stated objectives.

In implementing what has been termed Unit Planning -- Phase I, the decision was made to use existing historical data on enrollment, graduation, enrollment analyses for various units, etc. It was at this point that the author ex-

perienced new insights into human behavior regarding the acceptance or rejection of institutional data. Even though the data incorporated had been in existence for a number of years and had been distributed on many occasions unit administrators had expressed little concern about the accuracy of the data until it was incorporated into the planning process.

Simultaneously the initial steps were taken to formalize the organization for review of the existing information system and development of a model which would serve as a framework for implementation of the revised information system to be developed.

In our conference presentation, the authors will attempt to describe in more detail the highlights of pursuing these two objectives simultaneously and the interaction with various units of the institution, and in particular the relationship between the Office of Institutional Planning and the Computer Center. However for purposes of the paper itself the fundamental observation to be made was that the approach taken at this Institution and the progress made thus far would never have been achieved had it not been for the cooperation and resourcefulness of the administrators in the data processing area.

Why are they so valuable? First of all, at least in small to moderate size institutions individuals in these roles because of the nature of their work have not only an institutional view, but a great deal of experience and knowledge in working with all units of the institution. This institutional view and wide institutional experience places the data processing administrator in the position to respond to data base issues and procedures that go across the traditional boundaries of academic affairs, student affairs, administration, etc. Because they have been involved in the detailed planning of registration procedures, student billing systems, accounts payable, employee master file, as well as budget planning and budget reports they are in a position to view issues in an inter-related way more so than other administrators.

As a consequent, when the final decision was made on the organizational structure to review the management information system, the recommendation was made by the Office of Institutional Planning and approved by the President that the director and assistant director of the Computer Center would provide the leadership to each of the nine working committees by chairing each committee. That decision represented a major commitment of their time for many months, but was agreed to readily when it was understood that they were the only individuals on the campus



**362** who could address the topic not only from an administrative and policy view, but also from the technical view. I would hasten to add that the committees themselves are organized to give full representation to faculty and administrative staff throughout the Institution.

Appointment of data processing administrators to chair such committees gave immediate visibility to the issue of the role of the Computer Center and its staff in establishing policy and procedures versus maintaining a pure service role in which they carry out the policies and procedures established by other units of the institution. This was a positive occurrence since it brought to the surface an underlying issue that has needed discussion for a long, long time. A Computer Center is totally a service unit of the institution and should not be in a position of establishing policies and procedures except those internal to its own operation. This does not detract in any way however, from the expertise and leadership capability of data processing administrators in an overall planning effort. They are indeed resourceful people and we must use them to their fullest potential.



## MAINTENANCE PROBLEMS: DO IT YOURSELF AND YOU MAY DO IT BETTER

Lawrence J. Molloy  
David B. Preston  
Wayne County Intermediate School District

As the number of remote devices, such as scanners and terminals, used by local districts to enter data into the computer at the Wayne County Intermediate School District grew, maintenance problems grew also. Response to trouble calls was slow and costs were going higher. By establishing an in-house maintenance and special projects department, costs were not raised, response time was lowered, and many unique projects were completed. This paper details the formation of this department, its operation and benefits, and provides suggestions for others who are considering this type of service.

### Background

Wayne County Intermediate School District, located in the Detroit, Michigan metropolitan area, is the largest of the State's 58 intermediate districts, serving 36 local school districts with 704 buildings. The constituent districts conduct classes at the elementary, secondary, and community college level for more than 557,000 students, the third largest school district population in the nation. The mission of the Intermediate is to promote student learning by providing services to local school districts which will improve instructional programs and school district operations. Some of the services offered include in-service teacher training, curricula evaluation and modernization, occupational and career education, consultation services, research, and instructional and administrative data processing services.

In 1973 the Intermediate acquired an IBM 370/158 computer to provide services to the constituent districts. As it is the Intermediate's philosophy to offer as much flexibility as possible to the local districts, it was decided that all data entry functions should be done at the local site in an Remote Job Entry (RJE) environment. While keypunch equipment was adequate for many tasks, it was decided that optical scanning was more appropriate for the entry of large amounts of data. After examination of several options, the OP-SCAN 17 was selected as the standard scanner for use by the districts. Soon 20 of these devices were installed in local school districts. And a short time after that the first problems with maintenance

began to occur.

The problems that developed were service rather than equipment related. The hardware functioned as does most hardware. But the large number of devices in a concentrated area increased the frequency of problems that were reported to the Intermediate. The service personnel had too large a territory and too many devices to handle the calls at a rate that the Intermediate deemed acceptable. While some jobs may be delayed, others, like grade-reporting, must be done on time. Unfortunately, the response time for repairs did not seem to take these instant operations into account. Then, in the middle of discussions as to how to solve the maintenance problems, a new factor was introduced. A proposal was made to raise maintenance rates.

### An Idea Is Born

Faced with increasing costs, increasing delays in response to repair requests, and increasing number of devices, it was decided that other maintenance solutions should be examined. A study revealed that with the then current volume it could be cost effective to hire one service person for the Intermediate, purchase some tools and parts, and end the commercial maintenance agreement. This was accomplished in 1976 and the maintenance and special projects division at the Intermediate's Data Processing Department was formed. At its beginning it was decided to concentrate efforts on repairing devices that were widely used by the constituent districts. Initially,

**364** repairs were done only on OP-SCAN 17's and several DECwriter II's. However, as new users were added to the system and as user demands for new applications increased, the department was forced to expand its operations. The acquisition of several Hewlett-Packard mini computers for instructional applications not only increased the demand for DECwriter service but also introduced a demand for acoustic coupler repair. The implementation of a computerized guidance package, the Michigan Occupational Information System (MOIS), added 50 Cathode Ray tubes, 50 printers, and 50 couplers to the department's responsibility. Telephone rate increases caused several users to move into a multiplexing environment. This created a new service need for the repair of modems and multiplexors. Each new idea seemed to have its unique demands upon maintenance.

Throughout the department's growth period, one thought has been central to the Intermediate's maintenance efforts. Its primary responsibility is to respond to local districts' service needs in as little time as possible. Thus, new demands have created departmental growth. Maintenance is now staffed by two full time and two part time employees, as well as numerous summer and special project workers. Also, the need to keep the response time as low as possible has meant that certain devices are not repaired by the Intermediate's staff. For example, the staff is capable of repairing key-punches and teletypes but does not as it would strain their commitment to those users who pay for maintenance on the existing equipment. If a new idea has a large enough demand, such as an upcoming upgrade to the Intermediate-wide telecommunications network, additional resources are acquired to maintain the commitment to their existing service levels. Even maintaining this current equipment is sometimes difficult. This may be seen by examining the following list of equipment now serviced by the Intermediate's maintenance staff:

- 41 - Scanners (Op Scan 17)
- 53 - Hard Copy Terminals (DECwriter II)
- 60 - CRT's (Mostly ACT IV and ACT V)
- 50 - Printers (Integral Data Systems)
- 65 - Couplers
- 6 - Multiplexors (MICOM 720)
- 3 - Answer MODEMS (VADIC)
- 8 - Limited Distance Adaptors - MODEMS (Gandolf 404)

Considering the multitude of devices with the mystique of computers, the uniqueness of telephone lines, and Murphy's Law, it is obvious that the maintenance department is never out of work.

## A Philosophy For Service

Maintenance is a service function and, as such, is as much people related as machine related. The maintenance staff at the Wayne County Intermediate believe that their first priority is to get the user up as soon as possible. If, after diagnosing the problem at the user site, it appears that the problem can't be fixed within half an hour, every effort is made to replace the inoperable device immediately. In fact, on any CRT, printer, or coupler call, a spare is routinely taken just in case an exchange is necessary. On scanner calls a complete set of replacement boards is taken as insurance. This increases the probability that the user will experience minimum down time.

An important aspect of service is having the proper part at the right time. The Intermediate found it more economical to purchase two new units of each device initially and use them for spare parts than to purchase spare parts separately. In addition, certain parts with short life expectancy, such as rollers, are purchased in quantities directly from the manufacturers. Often an examination of a device may reveal a simple problem, such as a blown resistor or loose chip. These are fixed at the Intermediate. However, more complex problems, such as boards full of ROM, are returned to the manufacturer for repair. Experience has proven to be the best teacher as to what parts to stockpile for what machines. But a repair course, offered by many manufacturers for their devices, doesn't hurt either.

## A Typical Service Call

Perhaps the most important part of the service call is the initial conversation with the user. By carefully discussing the symptoms over the phone a good diagnosis may be possible. Wayne County Intermediate School District has found that 40% of its service calls can be handled over the phone. The solution may be to place a switch in the proper position, plug in a device, correct an operator error, or detect a communication or system problem.

If a service call is necessary, it is made as soon as possible. The Intermediate uses 'pagers' to reach maintenance personnel anywhere in the District. Once at the site, they attempt to operate the device in a test mode. If no problem occurs they have the user duplicate the conditions that created the problem. If no problem occurs during this test, it often indicates that the earlier problem was one of operator error. If a problem does occur, a diagnosis is made and attempts are made to repair the device.

Over 90% of the field problems are repaired at the user site in half an hour or less. Those that are not are solved by exchanging the unit for a replacement.

The field engineer log is an important last step in the repair process. The staff notes the time of the call, the time of response, the actions taken at the site, and the eventual disposition of the call. This provides a record that may be used as a historical record for each device. It also provides data for the recognition of maintenance patterns, operator difficulties, parts recommendations, etc..

#### A Well Received Bonus

One advantage of adding maintenance personnel to the Intermediate's staff was that it added a new perspective to problem solving. Their broad background in electronics allows them to propose solutions that might never be formulated by those more interested in software than hardware. These proposals often involved the engineering of special devices. Such efforts changed their role from maintenance to special engineering and research and development. Many special projects have been developed in this role. One project was the development of a Dual Terminal Adaptor (DTA) that linked the scanners to a hard copy terminal. The commercial version of this board is over \$1,000 while the DTA produced by the Intermediate is available to the local sites for \$150. A commercial box that switched a terminal from system to system costs \$500. The version produced by the Intermediate is available to the users for \$25. Cost reductions are not the only benefits from this special engineering service. Line drivers for long distance local communication have been developed. Micro computers are being examined as interfaces for systems work. In-house computerized typesetting is also being investigated. The special projects aspect of the maintenance department has become an important part of the Intermediate.

#### Does It Work?

The operation of an in-house maintenance and special projects department at the Wayne County Intermediate School District has been a success. It has kept user down time to a minimum. It has responded to changing user needs. It has developed special devices that may not be otherwise available or are available only at a dramatic increase in cost. Finally, it has held costs constant for its users. The costs for maintenance have remained the same as those originally charged by the commercial vendor five years ago.

#### How Do I Get Started?

365

Each area has its own unique needs. But the following guidelines should be helpful for anyone considering developing an in-house maintenance and special projects department:

1. Identify the devices you wish to maintain and concentrate on them; try to be very selective at the start.
2. Hire someone who knows how to repair those devices or send staff to a reputable school for training.
3. When selecting staff look for individuals who:
  - a. like people. Over 60% of their job will involve working with people and even the best knowledge of hardware can't offset that responsibility;
  - b. have had experience in servicing electronic equipment;
  - c. have an educational background that includes a variety of courses in electronics.
4. Never forget that maintenance is a service function.
5. Give the department adequate space for bench work and adequate budget for acquisition of parts.
6. Acquire the necessary tools. For less than \$2,000 one can purchase the following minimal starting tools:
  - a. an Oscilloscope (\$1200)
  - b. a Voltmeter (\$125)
  - c. a Field Engineering Tool Kit (\$350)
  - d. a Break-Out Box to Monitor Communications on EIA Lines (\$225)
7. When attending manufacturer's schools spend as much time as possible making contacts with staff in their customer service, shipping, and parts departments. These contacts may come in handy when a part is needed.

#### Conclusion

An in-house maintenance and special projects department can be an asset to any organization. Depending upon the

**366** size of the organization it may well reduce maintenance costs. It will reduce down-time and offer new opportunities to experiment with hardware. As one systems analyst at Wayne County Intermediate School District said, "I had never heard of an in-house maintenance department and now I don't know how I worked without one."



## A MARKET NEEDS APPROACH TO THE SYSTEMATIC SELECTION OF AN ECONOMICALLY EFFICIENT EDUCATIONAL COMPUTER RESOURCES SYSTEM

Dr. Joseph L. Braden, Associate Professor  
Dr. Asad S. Khailany, Associate Professor  
Eastern Michigan University

Determination of computer resource requirements is a complex task in which a systematic problem solving approach must be used. A computer resource system is made up of hardware, software and people. Needs in these areas must be determined for the existing computer system, projected use in twelve months, and projected use in five years.

The analysis approach can be divided into a four step process. First, the specific uses to which the system will be put must be specified for all three time frames. Second, the existing computer system must be evaluated. Third, system changes needed to meet present, twelve month, and five year needs are specified. Fourth, potential vendors are contacted for availability and price information.

### Introduction

The type of computer resource system needed by an organization depends on the nature of the organization and its needs. The following steps summarize a systematic approach to determining the system goals of the organization and the type of computing resources which must be acquired by the organization to achieve its goals.

To complete the computer resource system selection, the organizational goals are defined for three time periods, the hardware and software necessary to meet the goals are determined, the staff needed to manage and run the system are specified, and the physical requirements for these to operate properly are defined.

At this point the proposed system is compared with the existing system by looking at costs and benefits.

Computer resources are made up of hardware (the electronic equipment and machinery), software (the programs that make the hardware do what people want done), and people (who determine what is to be done, write the software, and handle getting programs and data into and out of the hardware).

Computer resources can be applied to administrative tasks, research projects, evaluation projects, computer-related courses, and computer aided instruction (CAI).

The first step in determining computer resource requirements is to decide on the specific uses to which the system will be put. This projection must be made for immediate needs, needs expected to occur within the next twelve months, and needs that will probably exist up to five years ahead. This must be completed for all possible areas that will be using the computing system.

It may be desirable to identify and visit places having similar needs that have an existing system installed.

The second step is to evaluate the existing computer system in detail. The third step is to specify system changes required to meet twelve month needs, and to meet five year needs. The final step is to contact potential vendors for availability and price information.

At this point selection of the computer resource system is determined by available funding. If the five year needs system cannot be purchased, the system obtained must allow for expansion when additional funds become available.

### Determining Computer Resource Requirements

Numerous items must be taken into consideration when completing these four steps. The following is a list of such items, to select a system to achieve the specific goals within the time frames:

- The maximum number of users
- The system type needed
  - batch
  - timesharing
  - remote job entry
  - combination
- The compiler(s) needed
  - basic
  - cobol
  - fortran
  - assembler
  - others
- The type of interaction needed
  - cards
  - CRT
  - punched paper tape
  - magnetic tape
  - printer(s)
- System software
  - operating system
  - utility programs
  - application packages
- Other services needed

#### Computer System Hardware:

- Computer
  - main memory
    - type
    - memory size
    - speed
    - byte or word
  - interfacing
- Storage devices
  - tape unit(s)
  - disk unit(s)
    - size
    - speed
  - diskette
  - other secondary storage devices
  - punched paper tape
  - card punch
- Peripheral devices
  - line printer(s)
  - card reader(s)
  - terminal(s)
  - key punch(s)
  - keyboard to tape
  - keyboard to mag card
  - keyboard to diskette
  - telephone transmission controller
  - plotter
- Interconnecting cables

At this point, contact vendors to obtain the following information about their available equipment meeting the needs stated above, for all time frames:

- Product specifications
- Model
- Purchase cost
- Lease/option to buy cost
- Lease cost
- Rental cost
- Physical and electrical requirements for products

#### Systems, Compilers and Utility Programs Available from the Vendors:

- Software
  - operating system

- batch
- timesharing
- mixture
- compilers available
  - basic
  - cobol
  - fortran
  - others
- utility programs

For these items, determine what physical requirements are needed, such as:

- room(s)
- desks
- seats
- telephones
- temperature control
- humidity control

-For software, determine if it is included in a package price, or if not, the cost to buy and the cost to implement and maintain it.

Another very serious consideration is that of application programs, existing and needed in the future. Existing must be transferred to any new system with little or no problems. This requires investigation of such things as:

- Character codes
- Input medium and data
- Output medium and data
- Program language
- Program size
- Program requirements
- Operating system environment
- Cost of modifications
- Maintenance cost
- Program purpose
- Storage requirements

This investigation must also look at packages, and in addition look at:

- Source
- Package purchase cost
- Training cost
- Reports produced
- Copyright law requirements

Now the computing system staffing requirements must be projected. This will include management, technical, clerical and organizational considerations. The basic organization types are:

- Centralized
- Decentralized
- Distributed

Managerial lines of authority must be determined:

- Report to superintendent
- Report to assistant superintendent of:
  - academic affairs
  - administration
  - business affairs

Job specifications for all positions must include:

- Job title
- Work to be performed

- Academic qualifications
- Experience qualifications
- Salary
- Fringe benefits

Finally, financial and physical requirements must be collected. The climate and space needs can easily be estimated once all the above information has been determined. In addition, the following must be considered:

- Costs of:
  - backup facilities
  - security
  - insurance
  - utilities
    - heat
    - cooling
    - humidity control
  - fire protection
  - electrical failure/surge protection
  - telephone
- Connections between areas
- Physical items
  - raised floors
  - work space
    - desks
    - seats
  - storage space
- Maintenance contracts
- Annual supplies
  - type
  - sources
  - schedule
  - cost

### Conclusion

The approach taken here looks at the selection of a computer resource system from the point of view of the ultimate user. The organization goals must be the first item investigated if the computing system is to economically assist the organization in reaching its goals.

## TITLE INDEX

|                                                                                                                                                             |     |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| AUMS: A Score Language for Computer-Assisted Applications in Music<br>~ W Kenton Bales & Joan C Groom . . . . .                                             | 47  |
| The Assessment of Students' Computer Literacy -- Results from the MECC<br>Computer Literacy Study ~ Anderson, Hansen, Johnson & Klassen . . . . .           | 175 |
| Automatic Programming: An Educator's Aid For Computer-Assisted Test<br>Construction (CATC) ~ Ronald W Collins & Steven J Duff . . . . .                     | 51  |
| Breaking Down the Barriers: The Real Micro Miracle ~ Darlene E Crawford . . . . .                                                                           | 177 |
| CAI Literacy in Instructional Technology Curriculum ~ Marsha Young . . . . .                                                                                | 181 |
| CAMIL and the Effective Implementation of Instructional Software for a<br>Large Scale CMI/CAI System ~ David Pflasterer & Anne Montgomery . . . . .         | 277 |
| The Case of an Educational Data Center's Environmental Transition: Its<br>Affect on People, Places, and Things ~ Philip A. Mackey . . . . .                 | 325 |
| Choosing a Classroom Microcomputer ~ Ed Herstein . . . . .                                                                                                  | 328 |
| Computer Access to Information: The Academic Library of Tomorrow<br>~ Robert R Korfhage & Christine L Borgman . . . . .                                     | 58  |
| Computer Assisted Test Construction at the University of Georgia<br>~ John S Edwards & James E Eisele . . . . .                                             | 62  |
| The Computer Attitude Continuum (CAC): Sorting the Critics and the<br>Appreciators ~ Walter M Mathews & Abraham W Wolf . . . . .                            | 66  |
| The Computer Based Hourly Classroom Attendance System in Kalamazoo Public<br>Schools ~ Jerry Pochtman . . . . .                                             | 235 |
| Computer-Based Instruction in Medical Education ~ William E Sorlie<br>& Diane L Essex . . . . .                                                             | 70  |
| The Computer in Cartoons: The Saturday Review, 1946-1976 ~ Kaye Reifers<br>& Walter M Mathews . . . . .                                                     | 184 |
| Computer Literacy for 4th, 5th & 6th Graders ~ Sam Ewing . . . . .                                                                                          | 188 |
| A Computer Literacy Module for Junior High School Students ~ Ione Goodman,<br>Steve Wetherbee, Daniel Neaton & John Marshak . . . . .                       | 191 |
| Computer Literacy, School Mathematics, and Problem Solving (3 for the price<br>of 1) ~ Fred Bell . . . . .                                                  | 194 |
| Computer Management of Educational Competency Statements ~ Fred G Smith<br>& John S Edwards . . . . .                                                       | 74  |
| A Computer Management System for a Modularized Course with Hierarchical<br>Structure and Optional Units ~ Ann Brebner, J.H. Hallworth & W.D. Winn . . . . . | 76  |



|                                                                                                                                                                    |     |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| Computers in the College Classroom: An Example from Transportation<br>Geography - C Murray Austin . . . . .                                                        | 81  |
| A Computer Supported Field Based Teacher Preparation Program<br>- John H Schnabel . . . . .                                                                        | 198 |
| Contrasts in Achievement of Comp Skills by Low and High Ability 6th Grade<br>Students in 1965 MSG & 1975 Modern Elem. Math Programs<br>- Ann D Hungerman . . . . . | 85  |
| CRSIE - A Computer Resource System for Instructional Evaluation<br>- Randy Rousse . . . . .                                                                        | 88  |
| CVIS: Where Career Education and Computer Meet - Dianne K Garnett<br>& Robert F Garnett . . . . .                                                                  | 91  |
| A Data Management System for a Small Liberal Arts College - Victor A Meyer<br>& Ignatia Frye . . . . .                                                             | 238 |
| Data Processing in the Year Round 45/15 Plan School District<br>- Donald J Vinlove . . . . .                                                                       | 244 |
| A Degree Progress Audit of Students by Computer - James E Greene<br>& Lester A Singletary . . . . .                                                                | 247 |
| Delaware Technical & Community College Student Record System-Why it is<br>Different - Frank A Kros III . . . . .                                                   | 252 |
| The Development of a Management Information System: A Case Study<br>- Edwin D Bell . . . . .                                                                       | 330 |
| The Dialectic of Artist and Technologist in Instructional Systems<br>Development - G.P. Kearsley & A.F. O'Neal . . . . .                                           | 333 |
| Documentation Standards and Their Application to Computer Systems<br>- Frank B Thomas . . . . .                                                                    | 336 |
| Educational Administration, Evaluation, Research and Package Programs: An<br>Integrated EDP Application - S James Zafirau . . . . .                                | 95  |
| Educational Programs in Computers and Society and Computer Literacy<br>- Richard H Austing & Gerald L Engel . . . . .                                              | 203 |
| The Effects of Computer-Assisted-Instruction on Low Achieving Math Students<br>in the Junior High School - Ione T Goodman . . . . .                                | 3   |
| Effects of Computer-Based Instruction and Tutoring in an Elementary School<br>System - Lotus M Knief . . . . .                                                     | 98  |
| The Effects of Supplementary CAI on the Math and Reading Achievement of EMR<br>High School Students - A Thomas Vincent . . . . .                                   | 6   |
| Elementary Computers - James Hollingsworth . . . . .                                                                                                               | 102 |
| English Strands: A Status Report - Bruce G Alcock, David Nicholson<br>& Starr Snead . . . . .                                                                      | 12  |
| English Usage Exercises II: A CAI Series for the Elementary School<br>- John G Allee & Peter J Quinn . . . . .                                                     | 15  |
| Establishing a CAI Center and Developing Software: Problems, Concerns, and<br>Recommendations - Thomas S Jordan . . . . .                                          | 341 |
| Experiences of "My Pal the Computer" in the Primary Grades - Jean Rice<br>& Sandy O'Connor . . . . .                                                               | 210 |
| Exploring the Potential Use of Computer Assisted Instruction in a Nursing<br>Curriculum - Patricia J Morin & Julia Ann Fisco . . . . .                             | 19  |

|                                                                                                                                                                            |     |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| File-Management and Formating Techniques for an APL Test Generation System<br>- Daniel R Lawrence . . . . .                                                                | 104 |
| Florida's Strategy for Optimizing Computing Resources for Educational<br>Administration and Instruction - M D Roblyer & David Brittain . . . . .                           | 346 |
| From Timesharing to Microcomputers - Donald C Holznagel . . . . .                                                                                                          | 351 |
| A Generalized Management System for Educational Data - John S Edwards<br>& Pier Paolo Pisani . . . . .                                                                     | 256 |
| Humanism for the Computer Age in Education: Planning for Man/Machine<br>Synchronicity - George Donoian . . . . .                                                           | 21  |
| Humanistic Technology: Toward a Renaissance for Education - D Thomas King . . .                                                                                            | 283 |
| Implementation of the Apple II Microcomputer in Minnesota - Linda Borry<br>& John Arneson . . . . .                                                                        | 356 |
| An In-Service Model to Prepare All Elementary Teachers in One School<br>District to Use Computers - Gerard J Putz . . . . .                                                | 213 |
| Institutional Planners and Computer Center Administrators: Need for Greater<br>Dialogue - Robert F Garnett & Dianne Garnett . . . . .                                      | 359 |
| Instructional and Research Considerations with an Intelligent Videodisc<br>- Bobby Brown, Lawrence Stolurow, Barbara Fowler & Joan Sustik . . . . .                        | 287 |
| Instructional Decision Support Systems: Design Considerations<br>- William C. Bozeman . . . . .                                                                            | 109 |
| Integrating Computer Concepts into Community College Business Curricula<br>- Roy James Parrish, Jr. . . . .                                                                | 215 |
| The Interaction of Heuristic/Algorithmic Instructions and Conceptual<br>Preference in Computer Based Problem Solving - Joan M Sustik<br>& Bobby R Brown . . . . .          | 114 |
| Interactive Computer-Based Exercises in Radiology - Kreps, Kraft, Burris<br>& Olson . . . . .                                                                              | 119 |
| An Interdisciplinary Approach to Computer Science - William W Agresti . . . .                                                                                              | 219 |
| The Introduction of CAI to an Undergraduate Decision Making Course<br>- Susan S Lenker & Edward J Fisher . . . . .                                                         | 222 |
| Logical Distributed Data Processing for School Districts - S M Raucher . . . .                                                                                             | 291 |
| Magnet Math - Drawing the Best from Your Students by Computer<br>- Betty Vanden Bosch . . . . .                                                                            | 27  |
| Maintenance Problems: Do It Yourself And You May Do It Better<br>- Lawrence J Molloy & David B Preston . . . . .                                                           | 363 |
| A Market Needs Approach to the Systematic Selection of an Economically<br>Efficient Educational Computer Resources System - Joseph L Braden<br>& Asad S Khailany . . . . . | 367 |
| Micro in a Macro World: Curriculum Development for the Microcomputer<br>- Shelley Yorke Rose . . . . .                                                                     | 33  |
| The Microprocessor as Pencil - John D'Angelo . . . . .                                                                                                                     | 295 |
| "Universite du Quebec" PLATO System - Paul-Andre Desjardins . . . . .                                                                                                      | 37  |
| Projecting Enrollments for Elementary and Secondary Schools<br>- William J Hierstein . . . . .                                                                             | 259 |

|                                                                                                                                                                                     |     |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| The Projects with Industry Approach to Training and Placing Disabled Students as Computer Programmers - Robert J Leneway . . . . .                                                  | 225 |
| Random Access Audio Hardware, Software and Courseware - Donald H McClain . . . . .                                                                                                  | 308 |
| Random Generator for Arithmetic Expression Problems - James D Jensen . . . . .                                                                                                      | 125 |
| SCORE: A Test Scoring and Analysis Program - Edwin J Jaufmann & Robert L Williams . . . . .                                                                                         | 262 |
| Selecting Microcomputers for the Classroom - David B Thomas & Donald H McClain . . . . .                                                                                            | 312 |
| The Significance of Uncontrolled Variables on Research Outcomes of Computer Assisted Instruction - J F Huntington . . . . .                                                         | 127 |
| A Systematic Information Management Model for Cooperative Education: Data Base Development & Evaluation - Robin D. Avery & Jeanne Dulaney-Sorochak . . . . .                        | 266 |
| The Taming of a MINIMICROMAX - Leighton A Price & Raymond J Kalush, Jr . . . . .                                                                                                    | 130 |
| Teachers Taking a Byte from an Apple - Earl Keyser . . . . .                                                                                                                        | 319 |
| There are no Blackboards - Ronald V Carswell . . . . .                                                                                                                              | 228 |
| The T.I.E.S. Instructional Management Support System - Donald Holznagel & James Sydow . . . . .                                                                                     | 136 |
| The TRS-80 as a Classroom Teaching Tool: Teaching Hebrew and Religious Studies - Samuel W Spero . . . . .                                                                           | 140 |
| Use of Information Retrieval Programs from A (IDS to Job Hunting) to Z (EBRA Keys) - Janice M. Glime . . . . .                                                                      | 144 |
| Use of the Personal Computer as an In-class Teaching Tool in Introductory Business Statistics - Sidney P Sytsma . . . . .                                                           | 150 |
| Using a Computer to Aid in Evaluating the Occup. Activities of a Student's Progress in a Physician's Assistant Program - Edwards, Gallagher, Valgora, Campbell & Costello . . . . . | 270 |
| Using the Computer to Foster Creative Interaction Among Students - Sharon Dugdale . . . . .                                                                                         | 153 |
| Uses of Computer Simulations in Introductory Psychology - Douglas B Eamon & William N Hayes . . . . .                                                                               | 160 |
| Utilizing the Computer to Teach and Study Clinical Problem Solving Skills - Doron Gil, Chris Wagner & John Vinsonhaler . . . . .                                                    | 164 |
| "We Deliver" (Jobs) - Robert J Miller, Gerri Powell-Rae & Roland L Meade . . . . .                                                                                                  | 169 |
| The Word-Problem Strand in Math Drill Using a Mini-System - G Lawrence Hopkins . . . . .                                                                                            | 39  |

# AUTHOR INDEX

| Author                   | Address                                                | Page        |
|--------------------------|--------------------------------------------------------|-------------|
| Agresti, William W       | University of Michigan-Dearborn, Dearborn, Mich        | 219         |
| Alcock, Bruce G          | Riverdale Country School                               | 12          |
| Allee, John G            | George Washington University                           | 15          |
| Anderson, Ronald E       | University of Minnesota                                | 175         |
| Arneson, John            | Minnesota Educational Computing Consortium             | 356         |
| Austin, C Murray         | University of Northern Iowa                            | 81          |
| Austing, Richard H       | University of Maryland                                 | 203         |
| Avery, Robin D           | Macomb County Community College, Mt. Clemens, Mich     | 266         |
| Bales, W Kenton          | North Texas State University                           | 47          |
| Bell, Edwin D            | North Carolian A & T State University                  | 330         |
| Bell, Fred               | University of Pittsburgh                               | 194         |
| Borgman, Christine L     | Chilton Corporation                                    | 58          |
| Borrry, Linda            | Minnesota Educational Computing Consortium             | 356         |
| Bozeman, William C       | The University of Iowa                                 | 109         |
| Braden, Joseph L         | Eastern Michigan University, Ypsilanti, Mich           | 367         |
| Brebnier, Ann            | The University of Calgary, Calgary, Alberta            | 76          |
| Brittain, David          | Florida Department of Education                        | 346         |
| Brown, Bobby R           | University of Iowa                                     | 114, 287    |
| Burris, Russell          | University of Minnesota                                | 119         |
| Campbell, Leonard        | University of Nebraska, Lincoln, Nebraska              | 270         |
| Carswell, Ronald V       | Texas State Technical Institute                        | 228         |
| Collins, Ronald W        | Eastern Michigan University, Ypsilanti, Mich           | 51          |
| Costello, D F            | University of Nebraska, Lincoln, Nebraska              | 270         |
| Crawford, Darlene E      | Independent School District 196, Rosemount, Minn       | 177         |
| D'Angelo, John           | Texas Instruments, Inc.                                | 295         |
| Desjardins, Paul-Andre   | University of Quebec                                   | 37          |
| Donoian, George          | Detroit Public Schools, Detroit, Mich                  | 21          |
| Duff, Steven J           | Eastern Michigan University, Ypsilanti, Mich           | 51          |
| Dugdale, Sharon          | University of Illinois                                 | 153         |
| Dulaney-Sorochak, Jeanne | Macomb County Community College, Mt. Clemens, Mich     | 266         |
| Eamon, Douglas B         | Albion College, Albion, Mich                           | 160         |
| Edwards, Jesse           | University of Nebraska, Omaha, Nebraska                | 270         |
| Edwards, John S          | Division of Educational Support, University of Georgia | 62, 74, 256 |
| Eisele, James E          | University of Georgia                                  | 62          |
| Engel, Gerald L          | Old Dominion University                                | 203         |
| Essex, Diane L           | University of Illinois, Urbana-Champaign               | 70          |
| Ewing, Sam               | Adrian Public Schools, Adrian, Mich                    | 188         |
| Fisco, Julia Ann         | University of Nebraska College of Nursing              | 19          |
| Fisher, Edward J         | Central Michigan University, Mt. Pleasant, Mich        | 222         |
| Fochtman, Jerry          | Kalamazoo Public Schools, Kalamazoo, Mich              | 235         |
| Fowler, Barbara          | University of Iowa                                     | 287         |
| Frye, Ignatia            | Marygrove College, Detroit, Mich                       | 238         |
| Gallagher, Thomas        | University of Nebraska, Omaha, Nebraska                | 270         |
| Garnett, Dianne K        | University of Evansville, Evansville, Indiana          | 91, 359     |
| Garnett, Robert F        | University of Evansville, Evansville, Indiana          | 91, 359     |
| Gil, Doron               | Michigan State University, East Lansing, Mich          | 164         |
| Glime, Janice M          | Michigan Technological University, Houghton, Mich      | 144         |
| Goodman, Ione T          | Ann Arbor Public Schools, Ann Arbor, Mich              | 3, 191      |
| Greene, James E, Jr.     | Georgia State University                               | 247         |
| Groom, Joan C            | North Texas State University                           | 47          |



| Author                  | Address                                                      | Page     |
|-------------------------|--------------------------------------------------------------|----------|
| Hallworth, J H          | The University of Calgary, Calgary, Alberta                  | 76       |
| Hansen, Thomas P        | Minnesota Educational Computing Consortium                   | 175      |
| Hayes, William N        | Albion College, Albion, Mich                                 | 160      |
| Herstein, Ed            | Community High School, Ann Arbor, Mich                       | 328      |
| Hierstein, William J    | The University of Iowa                                       | 259      |
| Hollingsworth, James    | L'Anse Creuse Public Schools, Mt Clemens, Mich               | 102      |
| Holznagel, Donald C     | TIES, St. Paul, Minn                                         | 136, 351 |
| Hopkins, G Lawrence     | The Masters School, Dobbs Ferry, New York                    | 39       |
| Hungerman, Ann D        | The University of Michigan, Ann Arbor, Mich                  | 85       |
| Huntington, J F         | Miami University                                             | 127      |
| Jaufmann, Edwin J       | Corporate Time-Sharing Services, Inc.                        | 262      |
| Jensen, James D         | Rochester Institute of Technology, Rochester, New York       | 125      |
| Johnson, David C        | University of London                                         | 175      |
| Jordan, Thomas S        | Cleveland State University, Cleveland, Ohio                  | 341      |
| Kalush, Raymond J, Jr.  | Michigan State University, East Lansing, Mich                | 130      |
| Kearsley, G P           | Courseware, Inc.                                             | 333      |
| Keyser, Earl            | Northern Trails Area Education Agency, Clear Lake, Iowa      | 319      |
| Khailany, Asad S        | Eastern Michigan University, Ypsilanti, Mich                 | 367      |
| King, D Thomas          | St. Paul Public Schools, St. Paul, Minn                      | 283      |
| Klassen, Daniel L       | Minnesota Educational Computing Consortium                   | 175      |
| Knief, Lotus M          | The University of Arizona                                    | 98       |
| Korfhage, Robert R      | Southern Methodist University                                | 58       |
| Kraft, Robert           | University of Minnesota                                      | 119      |
| Kreps, Jeffrey          | University of Minnesota                                      | 119      |
| Kross, Frank A, III     | Delaware Technical & Community College                       | 252      |
| Lawrence, Daniel R      | Rochester Institute of Technology, Rochester, New York       | 104      |
| Leneway, Robert J       | Michigan State Technical Institute and Rehabilitation Center | 225      |
| Lenker, Susan S         | Central Michigan University, Mt. Pleasant, Mich              | 222      |
| Mackey, Philip A        | Delaware Technical & Community College, Wilmington, Delaware | 325      |
| Marshak, John           | Ann Arbor Public Schools, Ann Arbor, Mich                    | 191      |
| Mathews, Walter M       | The University of Mississippi                                | 66, 184  |
| McClain, Donald H       | The University of Iowa, Iowa City, Iowa                      | 308, 312 |
| Meade, Roland L         | Washtenaw Intermediate School District, Ann Arbor, Mich      | 169      |
| Meyer, Victor A         | Marygrove College, Detroit, Mich                             | 238      |
| Miller, Robert J        | Washtenaw Area Youth Placement Service, Ann Arbor, Mich      | 169      |
| Molloy, Lawrence J      | Wayne County Intermediate School District, Wayne, Mich       | 363      |
| Montgomery, Anne        | McDonnell Douglas Corporation                                | 277      |
| Morin, Patricia J       | University of Nebraska College of Nursing                    | 19       |
| Neaton, Daniel          | Ann Arbor Public Schools, Ann Arbor, Mich                    | 191      |
| Nicholson, David        | Riverdale Country School                                     | 12       |
| O'Connor, Sandy         | Fridley Schools, Fridley, Minn                               | 210      |
| O'Neal, A F             | Courseware, Inc.                                             | 333      |
| Olson, Frederick        | St. Paul-Ramsey Hospital                                     | 119      |
| Parrish, Roy James, Jr. | Danville Community College, Danville, VA                     | 215      |
| Pflasterer, David       | McDonnell Douglas Corporation                                | 277      |
| Pisani, Pier Paolo      | QNOSIS, Inc.                                                 | 256      |
| Powell-Rae, Gerri       | Washtenaw Area Youth Placement Service, Ann Arbor, Mich      | 169      |
| Preston, David B        | Wayne County Intermediate School District, Wayne, Mich       | 363      |
| Price, Leighton A       | Michigan State University, East Lansing, Mich                | 130      |
| Putz, Gerard J          | Macomb Intermediate School District, Mt. Clemens, Mich       | 213      |
| Quinn, Peter J          | St. John's University, New York                              | 15       |
| Raucher, S M            | Montgomery County Public Schools, Rockville, Maryland        | 291      |
| Reifers, Kaye           | The University of Mississippi                                | 184      |
| Rice, Jean              | TIES, St Paul, Minn                                          | 210      |
| Roblyer, M D            | Florida Department of Education                              | 346      |
| Rose, Shelley Yorke     | Dallas Independent School District                           | 33       |
| Rousse, Randy           | Ingham Intermediate School District, Lansing, Mich           | 88       |
| Schnabel, John H        | Southern Illinois University, Edwardsville, Ill              | 198      |
| Singletary, Lester A    | Georgia State University                                     | 247      |
| Smith, Fred G           | College of Veterinary Medicine, University of Georgia        | 74       |
| Snead, Starr            | Greenwich Country Day School                                 | 12       |
| Sorlie, William E       | University of Illinois, Urbana-Champaign                     | 70       |
| Spero, Samuel W         | Institute for Computers in Jewish Life, Skokie, Ill          | 140      |
| Stolurow, Lawrence M    | University of Iowa                                           | 287      |
| Sustik, Joan M          | University of Iowa                                           | 114, 287 |
| Sydow, James            | TIES, St Paul, Minn                                          | 136      |

| Author              | Address                                                | Page    |
|---------------------|--------------------------------------------------------|---------|
| Sytsma, Sidney P    | Ferris State College, Big Rapids, Mich                 | 150     |
| Thomas, David B     | The University of Iowa                                 | 312     |
| Thomas, Frank B     | The University of Akron, Akron, Ohio                   | 336     |
| Valgora, Valgene    | University of Nebraska, Omaha, Nebraska                | 270     |
| Vanden Bosch, Betty | Flint Public Schools, Flint, Mich                      | 27      |
| Vincent, A Thomas   | Belmont High School, Dayton Ohio                       | 6       |
| Vinlove, Donald J   | Valley View School District, Romeoville, Ill           | 244     |
| Vinsonhaler, John F | Michigan State University, East Lansing, Mich          | 164     |
| Wagner, Chris C     | Michigan State University, East Lansing, Mich          | 164     |
| Wetherbee, Steve    | Ann Arbor Public Schools, Ann Arbor, Mich              | 191     |
| Williams, Robert L  | St. John's University, New York                        | 15, 262 |
| Winn, W D           | The University of Calgary, Calgary, Alberta            | 76      |
| Wolf, Abraham W     | The University of Mississippi                          | 66      |
| Young, Marsha       | Rochester Institute of Technology, Rochester, New York | 181     |